# REPORT DOCUMENTATION PAGE

**Form Approved OMB No. 0704-0188**



**1. AGENCY USE ONLY *(Leave Blank)***

**2. REPORT DATE**
January 2001

**3. REPORT TYPE AND DATES COVERED**
Final Report

**4. TITLE AND SUBTITLE**

Integrated, Hands-Free Control Suites for Maintenance Wearable Computers–VHIC

**5. FUNDING NUMBERS**

C: F33615-00-M-6053

**6. AUTHOR(S)**

Jeff Valiton, Scott Grigsby, Tim Choate

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

***SYTRONICS***, Incorporated
4433 Dayton-Xenia Road; Building 1
Dayton OH 45432

**8. PERFORMING ORGANIZATION REPORT NUMBER**

SYT-798-001

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

AFRL/HECP
(Attn: Gloria L. Calhoun)
2255 H Street, Bldg. 33, Room 333
Wright-Patterson AFB OH 45433

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12a. DISTRIBUTION/AVAILABILITY STATEMENT**

Approved for public release; distribution unlimited

**12b. DISTRIBUTION CODE**

A

**13. ABSTRACT *(Maximum 200 words)***

Report developed under SBIR contract for topic Final Report for the Integrated, Hands-Free Control Suites for Maintenance Wearable Computers–VHIC. ***SYTRONICS*** and UDRI applied significant experience and familiarity with the warfighter's needs in aircraft maintenance to the problem of having hands-free control of wearable computers so that they may access technical data easily and effectively. This report outlines the development of an operation prototype that is an innovative combination of head-tracking and speech recognition for effective and intuitive interfacing that was used by subject matter experts to evaluate the utility of the system. A simple two-controller approach uses voice for text-and-click entry and head movement, as well as speech, for pointing. A dampered throat microphone filters noise and a simple inertial tracker provides cursor movement.

**14. SUBJECT TERMS**

Speech Recognition; Hands-Free Control; Voice-Head Input Controller; Wearable Computers; Aircraft Maintenance; Electronic Maintenance Technical Orders; Head-Mounted Display; Glasses-Mounted Display.

**15. NUMBER OF PAGES**

82

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UNL |

NSN 7540-01-280-5500

Standard Form 298 (Rev 2-89)
Prescribed by ANSI Std 239-18
298-102

*SYTRONICS, Incorporated*
Dayton, Ohio

*"Test Systems; Human Systems; Modeling, Simulation, and Training"*

# FINAL REPORT

FOR THE

# INTEGRATED, HANDS-FREE CONTROL SUITES FOR MAINTENANCE WEARABLE COMPUTERS--VHIC

CONTRACT NO.: F33615-00-M-6053

CLIN 0001AE

PREPARED FOR:

AFRL/HECP
(Attn: Gloria L. Calhoun)
2255 H Street, Bldg. 33, Room 333
Wright-Patterson AFB OH 45433

AFRL/MLKT
(Attn: James David Box)
2310 Eighth Street, Bldg. 167
Wright-Patterson AFB OH 45433-7801

PREPARED BY:

***SYTRONICS, Incorporated***
4433 Dayton-Xenia Road
Building 1
Dayton OH 45432

19 January 2001

# FINAL REPORT

FOR THE

# INTEGRATED, HANDS-FREE CONTROL SUITES FOR MAINTENANCE WEARABLE COMPUTERS--VHIC

CONTRACT NO.: F33615-00-M-6053

CLIN 0001AE

*SYTRONICS* PROJECT NO.: 798

PREPARED FOR:

AFRL/HECP
(Attn: Gloria L. Calhoun)
2255 H Street, Bldg. 33, Room 333
Wright-Patterson AFB OH 45433

AFRL/MLKT
(Attn: James David Box)
2310 Eighth Street, Bldg. 167
Wright-Patterson AFB OH 45433-7801

PREPARED BY:


***SYTRONICS, Incorporated***
4433 Dayton-Xenia Road
Building 1
Dayton OH 45432


19 January 2001

PREPARED BY:

SCOTT S. GRIGSBY
Principal Investigator

APPROVED BY:

JEFFREY C. VALITON
Program Manager

DEBRA M. WINNER
Quality Assurance Manager

## TABLE OF CONTENTS

## TABLE OF CONTENTS (con't.)



## APPENDICES



## LIST OF TABLES

## LIST OF TABLES (con't.)



## LIST OF ILLUSTRATIONS

## REVISION HISTORY

| REV. | DATE | DESCRIPTION |
|---|---|---|
| A01 | 19 Jan 01 | Initial Release. |

## 1.0 INTRODUCTION

In this Final Technical Report for the Integrated, Hands-Free Control Suites for Maintenance Wearable Computers, hereafter referred to as the ***Voice-Head Input Controller (VHIC),*** we describe our technical efforts and results, feasibility conclusions, and anticipated approach for Phase II. We begin with a summary statement of the problem, followed by a description of our Phase I objectives and approach. These are summarized from the Phase I Proposal and included here to set the context for our subsequent technical discussions. Next, we describe our approach, accomplishments, results, and feasibility conclusions under each objective individually. Finally, we summarize our feasibility assessments and conclusions, and present our recommended approach for Phase II.

## 2.0 PROBLEM

Maintaining today's highly-sophisticated and complex aircraft is challenging for experienced technicians, let alone first-term airmen. The maintainer needs to thoroughly know the aircraft from the inside out. Moreover, maintenance technical information is often the most costly element of life-cycle support for Air Force weapon systems. Affordable weapon system support depends on technology that can bring down the cost of producing and updating maintenance technical information. Previous ***Air Force Research Laboratory (AFRL)*** research suggests that an automated maintenance aid that provides the maintainer with specialized technical information, presented in a format specifically designed to support the maintenance task, would significantly reduce maintenance costs (Thomas, 1995[a]). Such an aid would provide the technician with access to integrated information, including engineering drawings, flight tolerances, and complete ***Technical Orders (T.O.s)*** for the aircraft.

To this end, the Air Force has proposed an electronic T.O. system to replace the current paper-based system for future aircraft with the candidate application of ***Head-Mounted Display (HMD)*** devices and wearable computer systems. The Air Force has conducted research on mobile maintenance computing systems for over ten years, but the development of wearables for maintenance applications has, to date, focused on developing display system components, display formats, and display sequencing for technical data interfaces. During this time, system components such as displays and controls have each evolved to allow more mobility and less hands-on interaction. Yet, these advances in technology have not always proven to enhance maintenance performance. The key to improving maintainer performance is to approach the problem *from a human engineering perspective* in order to integrate maintenance hardware (displays, controls), software, personnel requirements, and work environment. Improvement to the system components cannot by itself improve task performance--the system must be usable and useful to the technician.

Display devices for mobile computing systems have evolved from hand-held displays to head-mounted monocular devices, and more recently to ***glasses-mounted displays (GMDs)*** (GMDs; see Figure 2.0-1). Advancement to the HMD allows for the technician's hands to be free. The GMD permits greater mobility around the aircraft and in tight places on the aircraft. In a study conducted by the ***University of Dayton Research Institute (UDRI)*** for AFRL/HESR (Kancler, Revels, Quill and Nemeth, 1999[b]), it was found that the GMD (and hence, wearable computers with HMDs, in general) should be used for tasks that require the user to switch attention between the GMD and the external environment. For example, a GMD type of system is more appropriate for a mobile maintenance application (where a maintainer's focus goes between the aircraft itself and the T.O. on the GMD) than it would be for an airline passenger writing a report while flying home from a business trip. Using these findings, a second study applied the GMD to a mobile computing environment (Revels, Kancler, Quill, and Nemeth, 1999[c]). This test employed a dual-task paradigm--a mobility task and a computer/GMD task. Both spatial and verbal screen formats were used for the computer task. In this study, spatial formats were found to be more compatible with the GMD than verbal formats. Allowing the users to easily switch attention to the external environment and providing them with spatially-oriented formats for their displays has great significance to integration and subsequent usability of computer control devices. That is, this research indicates that control devices should accommodate task attention switching and spatial display formats.

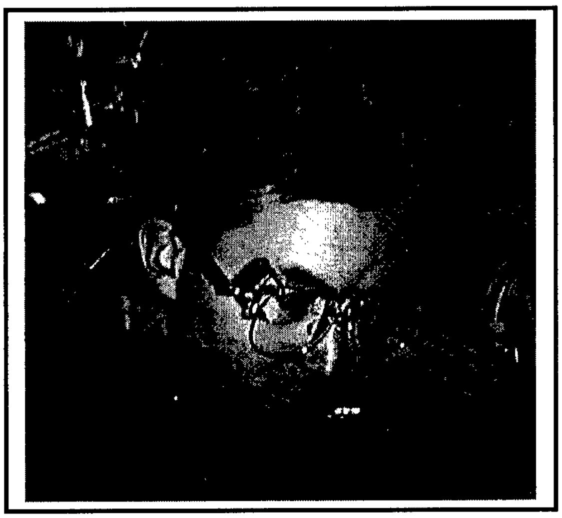


**Figure 2.0-1. Glasses-Mounted Display**

Regardless of the control implementation for a given wearable system under normal circumstances, the user must be able to provide several types of input. In the ***two-dimensional (2-D) Microsoft (MS)*** Windows operating environment, there are three primary types of input: (1) pointer movement--the positioning of the cursor on the screen (commonly performed by moving a mouse), (2) discrete input--the selection of an object of focus (e.g., the left mouse button click), and (3) text entry fill-in (e.g., standard keyboard entry). However, future interactive T.O.s may also include interactive ***three-dimensional (3-D)*** graphics. The presentation of 3-D objects on 2-D displays leads to questions of how users will interact with those objects (Stute, Bautsch, Gdowski, Calhoun, Grigsby, Dixon, Cunningham, and Stautberg, 1998[d]). To gain full benefit of 3-D object modeling, objects need to be manipulated in different ways such as translation, rotation about any axis, and size scaling. The need for a zooming (or size scaling) control is due to the limited resolution and screen size of the display. Certain aspects of a perspective drawing may need to be magnified or "exploded" in order to visualize individual parts, wires, or connections. Therefore, it is important to develop a system that can easily provide pointing, 3-D graphic manipulation, discrete input, and text entry in a reliable and intuitive way. Speech control integrated with a pointing device provides a very user-friendly and powerful 3-D graphic controller. Speech allows the user to easily change interaction mode so that the same controller or user action can provide many different manipulations. For example, the user could say "select" and then point to the 3-D object they wish to choose. Then they could say "zoom," now pointing to an object would magnify it. Next, they could change modes again by saying "rotate." Now, pointing in a direction would cause the object to rotate in that direction. Saying "translate" would cause the pointing action to move the object, etc. Combining speech with a single action can greatly increase the user interaction in an intuitive and simple yet very powerful manner.

Numerous research programs have focused on computer control devices for mobile computing. Since the early 1990s, control systems such as wrist-worn and voice controls have been researched. While these devices allow for mobility (e.g., wrist-worn keyboards and trackballs) and hands-free control (e.g., voice), they may not accommodate task attention switching and spatial displays. In fact, in many instances, the value of a hands-free control device could not be shown in a flight line setting (Chapman and Simmons, 1995[e]). Only recently has research been able to show the value of hands-free control for maintenance computing. In a ***SYTRONICS***/URDI study conducted for AFRL (McMillan, Calhoun, Masquelier, Grigsby, Quill, Kancler, Nemeth and Revels, 1999[f]), a synthetic maintenance environment was used where selected task characteristics were controlled. Two hands-free control suites--an alternative control suite and a voice-only control--were used along with a conventional manual control suite. The alternative control suite utilized different types of controllers for the three different input tasks: head-tracking for pointing, facial ***Electromyogram (EMG)*** (eyebrow lift or jaw clench) for discrete input, and voice for text entry. This mirrors the types of controllers required with conventional systems--mouse/trackball for pointing, button clicks for discrete, and keyboard for text entry. Because the voice-only system used speech alone which is difficult and clumsy to implement for pointing, the software had to be modified to eliminate the need for pointing. However, the alternative and voice-only systems resulted in equivalent performance on the entire maintenance task. In hands-busy portions of the task, the two hands-free control suites resulted in significantly better maintenance performance than the hands-on conventional control suite. However, once the suites were integrated into a flight line maintenance environment, performance improvements were diminished by usability issues such as: having to continually adjust the eyepiece to get a good field-of-view without interfering with line-of-sight to the task; having to work around the wires and cabling; using the wrong command words; inadvertent click (EMG) inputs so the user got "lost" in the software; etc. Some of these issues were directly related to use of the controls. However, others had to do with use of the system as a whole and show the importance of designing a fully-integrated, human-engineered system. Furthermore, the alternative system, though having superior performance in the lab, needed more natural/intuitive interfacing to be useful in the field. As one subject stated about the alternative suite, "It seemed okay when I was just here practicing on the computer, but when I actually got out there… it's just not as natural… you are thinking about something else instead of what you are trying to do." However, the voice control was called "very manageable, very usable… a cleaned up version would be great."

Speech recognition is currently available on some commercial wearable systems, but questions have arisen about its usability under high-noise and dynamically changing noise environments (Chapman and Simmons, 1995[e]). Current technological innovation in the area of noise filtering throat microphones has added to the application range of speech-based systems. When these microphones are used in conjunction with small vocabulary command/control functions and robust, tailored speech engines, the effects of noise can be severely curtailed, thus making speech an extremely viable option once more. However, the need for both verbal and spatial interaction precludes the use of speech alone. As alluded to above, speech is very poor at assigning direction and movement; therefore, another modality is required. The three primary candidates for off-system pointer movement (i.e., excluding trackballs, force stick, etc., which may be mounted on the ***Central Processing Unit (CPU)*** case or wrist-worn) are gesture, eye-tracking, and head-tracking. Gesture-based sensing usually utilizes sensors placed on the fingers, hands, or arms and, therefore, are not a good candidate for use during hands-busy tasks. Eye-tracking technology to date is costly, cumbersome, and entangled with numerous human engineering problems.

Attempts at eye-based control for wearables have produced mixed results. One advantage of using eye-tracking with wearables as opposed to desktop systems is that having both the display and eye-tracker head-borne precludes the need for head tracking to compute line-of-sight from the eye-gaze data. In this instance, eye gaze is line-of-sight. However, the use of eye-tracking systems with wearable computers and HMDs is still problematic. ***Electro-oculography (EOG)***-based systems measure the electrostatic field of the eye with skin electrodes placed near the orbit. These systems suffer from direct current drift and nonlinear output functions due to skin resistance changes and variations in the corneal-retinal potentials over time (Borah, 1989[g]) making absolute position sensing extremely difficult. However, this would not preclude their use as relative position changers where the user could look left to move the cursor left or look up and to the right to move the cursor up and to the right. This method is not as applicable for eye pointing as it is for head pointing because anything other than absolute positioning (i.e., cursor position within ~1 degree of foveal fixation) would force the user to acquire the target using peripheral vision. Other tracking methods rely on tracking of image-based features such as the corneal reflection (and/or other Purkinje images), light and dark pupil, limbus/scleral edge detection, etc. These systems can be faster to use than a conventional mouse when selecting moderate-sized targets; (Ware and Mikaelian, 1987[h]), however, speed savings on the order of tens of milliseconds are meaningless in a maintenance setting. While showing great promise in laboratory settings, these systems usually use IR sources and cameras which suffer marked performance degradation in changing ambient light conditions and would be saturated in direct sunlight unless extraordinary protection measures were taken. Furthermore, current systems are expensive, cumbersome, not easily incorporated into a HMD, require a large amount of processing, have an accuracy of only one degree or so, and would require calibration when donned and/or realigned or bumped. All of these problems can be avoided by the use of a low-cost inertial head-tracker. Furthermore, head-tracking is comparable to eye-tracking for large targets and better for smaller ones (Borah, 1995[i]).

This leaves head-tracking as a viable, cost-effective choice. Our studies and others have shown head-tracking to have performance equal to or better than conventional controls. Field studies have found it to be well received by maintainers. For example, a study comparing eye, head, and an eye/head hybrid control (Borah, 1995[i]), found that head control of cursor position is viable over a wide range of target sizes and motion distances. While not as fast as mouse control, it is comparable to pure eye control (for larger targets) and superior to an eye/head hybrid over most of the range. Furthermore, while it has been shown that a laser-based head-tracker was slower than touchscreen or ***Target Designation Control (TDC)*** on throttle (Liggett, Kustra, Reising and Hartsock, 1997[j]), these researchers concluded that these instances were due to problems with the equipment and not with head performance *per se.* They further concluded that "with further refinements, [head-tracking] has the potential of becoming as viable and intuitive as touch."

## The ***SYTRONICS*** Solution

After reviewing the problem and applying our past experience with designing and developing hands-free control systems, we postulated our proposed solution, the "Voice/Head Input Controller" or VHIC, as illustrated in Figure 2.0-2. VHIC utilizes a simple two-controller approach--voice for text and click entry, and head movement for pointing. A dampened throat microphone aids in noise filtering and a simple inertial tracker provides adequate cursor movement. This approach integrates the simplicity and ease of a speech-based system with a proven cost-effective pointing solution and avoids the problems wrought by eye-tracking complexities.

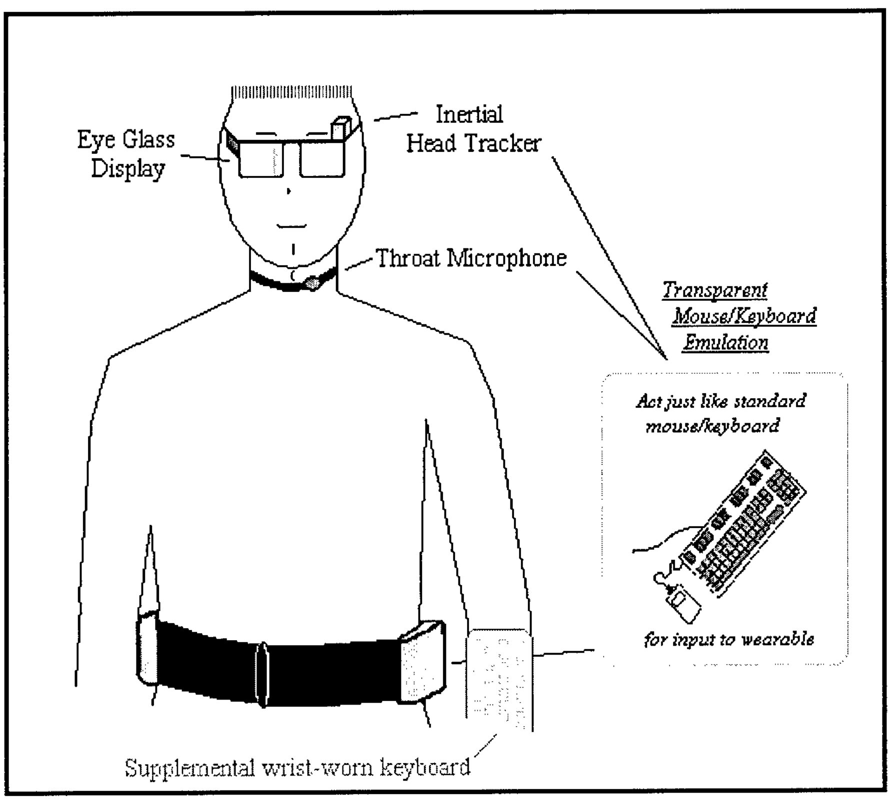



**Figure 2.0-2. The *SYTRONICS* VHIC Solution**

We believe, and our Phase I concept demonstration shows, that head pointing and speech control are the best candidates for providing a workable, innovative design for hands-free control of wearable systems. Both technologies have a long history and have matured to the state where they are operational, well-received, and cost-effective ( Anderson, 1998[k]; Borah, 1998[l]; McMillan, Eggleston and Anderson, 1997[m]).

## 3.0 PHASE I OBJECTIVES AND TASKING

### 3.1 Objectives

To accomplish the Phase I SBIR goals of establishing feasibility and preparing a solid foundation for Phase II, specific, realistic objectives were required. These objectives must also consider the specific problem issues we discussed in Section 2.0 as well as the need for transitioning the technology to a commercial product. In consideration of all these aspects, our Phase I Objectives were:

**Objective 1: Applicability and User Acceptance.** Under this objective, we sought to determine by literature review and by interviewing ***Subject Matter Experts (SMEs)*** such as maintainers, supply officers, logistics experts, and electronic T.O. developers, when and where the maintainers will be required to use spatial interaction (point and click, scrolling, resizing, etc.) versus large vocabulary verbal interaction (continuous text dictation). We also sought to determine what the user's preferred command/control vocabulary is, and how and where the proper nomenclature is routinely used when interacting with electronic T.O. formats such ***as Portable Document Format (PDF)*** and ***Interactive Electronic Technical Manuals (IETM)***.

**Objective 2: Ease-of-Integration into User Systems.** Under this objective, we sought to design a system that can work across all platforms and could even be retrofit to current systems. This would eliminate the need for alternative control system designers to have knowledge of, or have to wait for, the underlying code and layout of the future T.O. software or maintenance logistics systems which are still under development.

**Objective 3: Innovative Application of Proven Technology.** Under this objective, we sought to integrate a commercially-available continuous/discrete (command/control) speech recognition system with an inertial head-tracker and develop the mouse emulation software.

**Objective 4: Systematic Testing of all System Elements.** Under this objective, we sought to test all elements of the system by collecting empirical data on individual task parts, comparing and contrasting elements of the system, and testing the interaction of multiple elements in a controlled environment.

**Objective 5: Determination of Commercialization Potential.** Under this objective, we sought to interact with potential users and commercial partners to determine the commercial product potential of VHIC. We used information from these sources to ensure our Phase I-level designs and tests are focused upon commercial markets as well as satisfying USAF needs.

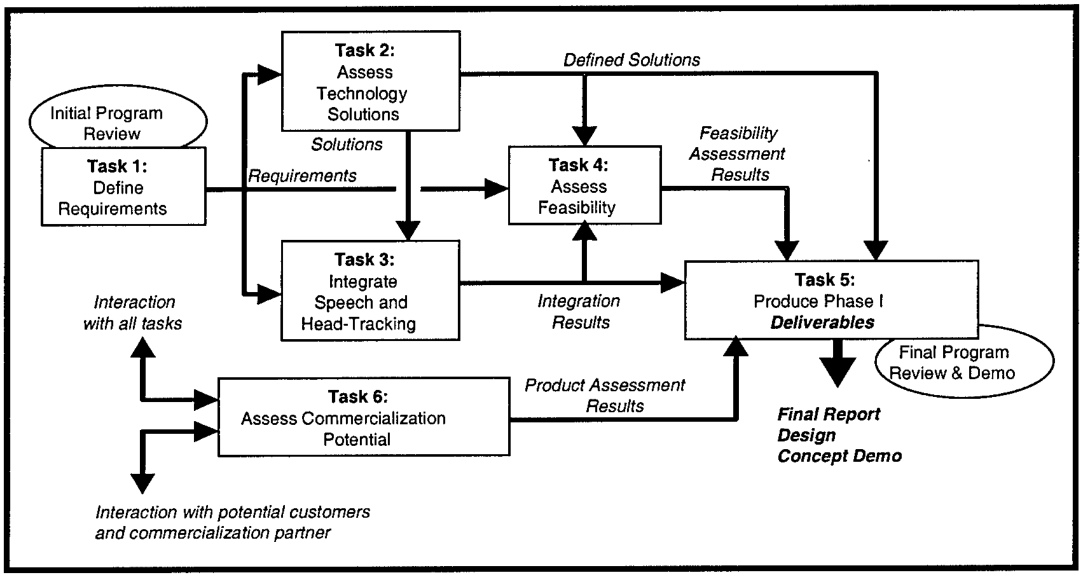



**Figure 3.1-1. Task Activities Chart**

## 3.2 Task Activities

Within the objectives outlined above, task activities were executed to: (1) achieve the Phase I objectives, (2) produce the Phase I products, and (3) form the foundation for Phase II. These task activites were:

Task 1: Define Requirements. Under this task, we interacted with maintenance organizations and our AFRL/HE sponsors to develop a realistic set of requirements for our Phase I activities. We realized that involving the maintenance organizations was critical to this effort and we also know that synergism with AFRL's goals was essential. We gathered the critical information to define requirements in our first task and conducted an ***Initial Program Review (IPR)*** under this task within the first month of the program to ensure complete project coordination and the correct approach to defining requirements. The results of this task yielded the requirements set for the other tasks.

Task 2: Assess Technology Solutions. Under this task, we determined the proper combination of speech-recognition products and head-tracking devices to provide the integrated mouse emulation we required under our concept for this effort. While suitable head-tracking devices are few and well-defined, there are myriad speech-recognition systems available and we needed to choose the one which satisfied our requirements to ensure a successful integration in the next task. We also assessed the state-of-the-art of wearable computer systems in order to determine the hardware constraints they placed on the system design.

Task 3: Integrate Speech Recognition and Head-Tracking. Under this task, we approached the integration from three perspectives--*design, implementation,* and *testing*. We *designed* the integrated system including speech, head-tracking, and additional software to achieve full mouse emulation control of a computer for maintenance purposes. We then *implemented* a test configuration consisting of selected head-tracking and voice-recognition systems, and prototype mouse emulation software written by ***SYTRONICS.*** We also *conducted an experiment* at AFRL/HE using actual electronic T.O. formats to ensure that our system met requirements. Here, we specifically addressed the issues in using head/speech techniques for mouse emulation. The products of this task were the designs, the test configuration, and the experimental results which form a solid foundation for Phase II.

Task 4: Assess Feasibility. In this task, we analytically assessed our designs from Task 3 and evaluated the results of our experiments (also from Task 3) by comparing these results with the requirements from Task 1. Through this comparison, we determined feasibility. The result of this comparison was a well-quantified assessment of feasibility, supported by our experimental findings, thus *satisfying the fundamental requirement of feasibility proof in Phase I.*

Task 5: Produce Phase I Deliverables. In this task, we produced the Phase I deliverables--a Final Report and a Concept Demonstration. This Final Report is the documentation of the actions, developments, discoveries, significant events, and results of all other Phase I tasks. It includes our Phase I-level designs and the feasibility assessments, as well as a preliminary definition of commercial product potential (from Task 6, described below). The Phase I Report fully documents all primary Phase I effort (six-month) accomplishments and results. The Concept Demonstration was the culmination of our experimental activities under Task 3 and demonstrated the feasibility results we obtained for AFRL/HE at AFRL. Our Concept Demonstration accompanied the Phase I primary effort--Final Program Review.

Task 6: Assess Commercialization Potential. In this accompanying task, we ensured that our Phase I efforts were focused toward a successful commercial product. Here, we share our Phase I accomplishments and results as they emerged with potential customers and commercialization partners.

In summary, our Phase I Objectives were reasonable, properly scoped for a Phase I investigation; and designed to address the problem, determine feasibility, and prepare for Phase II. Consequently, our tasking approach was designed to achieve these objectives. In the next several sections, we report on each of the Phase I Objectives individually discussing the approaches we used, our accomplishments, the results we obtained, and the feasibility conclusions we reached.

## 4.0 APPLICABILITY AND USER ACCEPTANCE--OBJECTIVE 1

In achieving this objective, we *answered the feasibility question of how best to design the system to provide the maintainers with an intuitive system that was functional and acceptable.* We also specifically addressed the critical problem issue of increased cognitive demand that can occur when users are forced to use unnatural or unfamiliar vocabularies and grammars to interact with the speech recognition system. As a result of achieving this objective, we formed the foundation for the development of the dual-mode speech recognition approach to handling high and dynamic noise levels for our Phase II effort.

Under this objective, we determined (by literature review and talking to SMEs) that the main formats currently being considered for electronic T.O.s are based on PDF (used with Adobe Acrobat Reader) and IETM. However, while no examples currently exist, "web page" based formats (such as ***Hyper-Text Markup Language (HTML)***) are also being considered for possible future deployment. We then obtained three examples of PDF formats (*F-16 Flight Control System Horizontal Stabilizer* [T.O. 1F-16AM-2-27JG-40-1]; *F-16 Flight Control System* [T.O. 1F-16A-2-27JG-00-2]; and *Installation of* ***Underbelly Protection System (UPS)*** *for the APR-46 Antenna on MC130H Aircraft* [T.O.- 1C-130(M)H-579]) and one IETM (a CD-ROM of *F110-GE-129 Engine data for the F-16*) to test when and where the maintainers will be required to use spatial interaction (point and click, scrolling, resizing, etc.) versus large vocabulary verbal interaction (continuous text dictation). Using these examples, we conducted a study to determine: what vocabulary the users preferred, what the command/control vocabulary is, and how and where the proper nomenclature is routinely used. The purpose of the study was to examine the combined use of voice commands and head-tracker as an alternative user interface for the digitized aircraft technical manual formats. This test utilized a hands-free interface comprising non-traditional or *alternative* control devices. As alternative controls, speech recognition and an inertial head-tracker were used as replacement devices for the keyboard and mouse, respectively. That is, speech recognition was implemented as a hands-free alternative to keyboard data entry (i.e., text) and mouse clicks. Similarly, the head-tracker was used as a hands-free alternative to the mouse for cursor placement.

## 4.1 Method

A two-staged approach was implemented for data collection. The purpose of the first stage was the collection and compilation of preferred, maintainer-specific voice commands when using the alternative control configuration. In this stage, subjects were asked to use the alternative control configuration, in whatever way necessary, to complete a series of instructions given by the experimenter. No pre-direction was given as to appropriate voice commands; rather, subjects were encouraged to use whatever voice commands they felt appropriate for a given action. The instructions were functionally based; that is, the intent of each instruction was to capture the method by which subjects utilized the alternative control configuration to complete a specific action, such as scrolling or highlighting. Experimenters recorded each subject's voice commands, cursor position, and overall user strategy. These data were compiled and a list of commonly-used verbal commands was generated.

The second stage of the study validated the voice command vocabulary through follow-up testing. This testing occurred approximately ten days after the initial test stage. Subjects were again given specific, functionally-based instructions by the experimenter. However, each subject was provided with a list of voice-commands to use during the session. The list was based upon the most commonly-used voice commands from the first stage of testing and was seen as a reasonable sample of voice commands. Subjects were asked to use the list as a reference, but also to report any voice commands that they felt should be added to the list.

4.2 Subjects

A total of 20 subjects volunteered to serve in this study. Aircraft maintainers from the Springfield, ***Ohio Air National Guard (OANG)*** F-16 facility and the 445$^{th}$ C-141 Air Force Reserve Unit from Wright-Patterson AFB served as subjects. For Stage 1, eight subjects from the OANG and six subjects from the 445$^{th}$ C-141 Air Force Reserve Unit participated. For Stage 2, three subjects from the OANG and three subjects from the 445$^{th}$ participated. Additionally, a retired "9-level" maintainer participated.

4.3 Apparatus

A laptop computer served as the CPU. Both electronic T.O. formats were loaded on this machine. One set of T.O.s was the IETM file developed by Lockheed for the F-16. The other set was a PDF file for the C-17. A 17-inch monitor served as the display device and was positioned directly in front of the subject at a viewing distance of approximately two to four feet. The head-tracking device comprised the internal hardware of a GyroPoint mouse mounted on a Kopin HMD (see Section 5.2 below for a description of the tracker). This configuration allowed the subject to move the on-screen pointer in direct proportion to lateral (pivoting left and right) head movements.

4.4 Variables

Because the intent of this study was to collect representative voice commands rather than performance levels, independent and dependent variables were not implemented in the classic experimental sense. Although two electronic T.O. formats were used, the study was not designed to measure performance differences elicited by these two formats. Rather, the development of a user vocabulary capable of addressing the functionality of both T.O. formats was the primary focus of the study.

4.5 Procedure

Each subject interacted with both electronic manual formats--PDF and IETM formats. Order was counterbalanced across subjects. It should be noted that the maintainer was not at any time working on an actual aircraft. Rather, the purpose was to collect information regarding the subject's strategies when interacting with computerized technical manuals via the hands-free input devices.

Before each session, subjects were familiarized with the head-tracking device, the capabilities of the software system they would be using, and the overall experimental procedure. Subjects were seated at a desk with the 17-inch monitor as the display device.

Two experimenters were involved with each experimental session. The first experimenter read a series of step-by-step instructions to the subject. The second experimenter played the role of voice recognition "software." The rationale for using a person to play the role of the voice recognition software instead of using real voice recognition software is multi-fold. First, the use of a human allowed the users to interact with the application in any manner they wished and still have it function correctly. If we had used real voice recognition software, the user's choices would be diminished because of the constraints of the software design. Since we sought to determine how the user wished to interact with the T.O.s, we decided to use an approach with more freedom. The use of a human instead of software also saved time by eliminating the need for training the voice recognition software and it also allowed the study to run in parallel with our efforts to develop the VHIC hardware and software and integrate the voice recognition software with the T.O.s.

Because a human was acting as the voice recognition software, a specific protocol was developed to capture the subject's voice commands for each step of the instructional set and to ensure that the voice commands were carried out correctly. The protocol was as follows-- Experimenter 1 would read the instruction (e.g., "Please go to the first warning."). The subject would then use the alternative control suite to carry out the command. Any voice commands were dealt with on an individual basis and involved Experimenter 2 temporarily taking control of the interface to carry out the command. For example, if the subject wished to use a voice command to scroll down to find the first warning, he might say, "scroll down." At this point, Experimenter 2 would verbally acknowledge receipt of the voice command by stating, "Got command." Experimenter 2 would then, via a separate input device, carry out the subject's instruction (in this example, he would click on the scroll down arrow once.) Once Experimenter 2 had completed the voice-commanded action, he would move the pointer to its location immediately prior to his taking control and return input control to the subject. The first experimenter recorded the cursor position and voice commands given by the subject. The subject and experimenters then verified whether the command was carried out as per the subject's expectations.

Each session was videotaped to ensure correct identification of cursor position and voice command vocabulary. An entire experimental session lasted from one to two hours. The result of Stage 1 was a list or "vocabulary" of common voice commands to be used during Stage 2. Also, specific user strategies were recorded with regard to the alternative control suite. These strategies were included as part of the recommendation section at the end of this summary.

Stage 2 of the experiment was essentially a replication of the first stage. The major difference involved the provision of a voice command list to the subject. Each list was specific to each electronic T.O. format, developed from the data collected in Stage 1, and sorted by functional action specific to the T.O. format. Where appropriate, the experimenters added voice commands to the list. To avoid duplication, a third list was developed containing functions common to both formats. For example, scrolling was a function common to both electronic formats. On this list, under the heading of *Scroll/Move* the following voice commands were listed:

1. Next Page/Previous Page
2. Scroll left/Scroll right
3. Scroll up/down
4. Page up/down
5. Continue Scrolling up/down (Stop)
6. Grab scroll (Stop)

Subjects were familiarized with the appropriate lists before starting the session and asked to refer to the list whenever necessary. However, subjects were also not required to use the commands on the list. In fact, subjects were asked, if it seemed appropriate, to add to the list.

## 4.6 Results

### 4.6.1 Summary of User Strategies

#### 4.6.1.1 Overview

User strategies seemed to be based upon on-screen interface characteristics. Generally, unlabeled icons and buttons elicited a point-and-click strategy, while labeled links elicited voice-only commands. Strategies also appeared to be based upon computer experience level. Users with increased familiarity with Windows-like interfaces seemed more prone to a point-and-click strategy, while novices seemed more likely to attempt various function-based voice commands. For example, when interacting with labeled hyperlinks, experienced computer users tended to move the cursor over the link and say, "select." Novice users, however, were more likely to attempt an alternative form of interaction, such as reciting the name of the link without using the cursor at all. User interaction can be described by the following three categories: (1) literal point-and-click translation; (2) point and use unique voice command; and (3) voice command only. A description of each strategy is provided below.

- *Literal point-and-click translation.* This strategy most closely resembles the interactive technique used with a mouse. This is considered a literal translation of mouse point-and-click because the pointer is used to indicate an item of interest and the click is activated by using a generic voice command. With such a strategy, the user moves the cursor to the target item (for example, a hyperlink to the index). The user then activates the target item via a voice command, such as "select" or "click." Users most familiar with a Windows-like interface seemed most likely to utilize this strategy.

- *Point and use unique voice command.* This strategy is similar to the literal point-and-click strategy, with the difference being the use of a context-specific voice command to activate the item of interest. In this strategy, if the user wants to activate the *index* hyperlink, he moves the cursor to the hyperlink and activates the link by saying the name of the link, "index." In this way, the voice command is unique to the target item.

- *Voice command only.* This strategy combines minimal use of the pointer with context-specific voice commands. For example, to access the *index* hyperlink via this strategy, the user simply says "index," without first orienting the pointer. Novice users, with little exposure to the point-and-click paradigm associated with Windows, seemed most likely to employ this strategy.

**NOTE:** *Use of "occupational vernacular."* While not a specific user strategy, the tendency to use voice command phraseology unique to aircraft maintenance was prevalent across subjects. For example, maintainers commonly refer to T.O.s by final portion of the hyphenated alphanumeric designation. This designation is commonly prefaced with the word "dash." Thus, the T.O. *B243-0001-123* would be referred to as "dash one two three."

## 4.7 Proposed Vocabulary

The vocabulary used for both formats is contained in Appendix A, "Task Data Summary." The subsections are sorted by function. Listed within each function are the voice commands and, where appropriate, cursor action. A frequency count is listed with each command representing the total number of times each command was used throughout testing.

## 4.8 Recommendations

Based upon the observed user strategies, several issues exist with regard to aircraft maintenance personnel and the use of the head-tracker and voice as primary input devices for a user interface. The user's experience level with mouse-driven interface formats seems to play a primary role in determining the user's technique with voice-driven input. The development of an acceptable voice recognition "vocabulary" should, of course, accommodate as many users as possible.

However, it should also be recognized that a voice recognition system offers capabilities not inherent to traditional point-and-click techniques. Most obvious is the capability to directly access a menu item, hyperlink, or functional capability without using a pointing device. The user vocabulary should support such capabilities, thus attempting to "break the mold" of the point-and-click paradigm. Realistically, however, there will likely be a transition phase between the point-and-click and voice-input user paradigms. This transition phase should also be accommodated by the user vocabulary. If such a transition phase is not accommodated, the user may never develop a paradigm which takes full advantage of the capabilities of voice recognition. Below are a series of recommendations as to the strategic application of a voice command vocabulary to a hands-free user interface.

Recommendation 1: Ensure that the user vocabulary will support multiple user strategies. Based on the observations from the present study, users tend to (1) combine the pointer and voice commands to mimic the traditional point-and-click strategies associated with a standard mouse, (2) utilize voice commands as a "direct manipulation" strategy, or (3) combine context-specific voice commands with cursor movement. The user vocabulary should support each of these strategies. Furthermore, the user may transition from one strategy to another, as he becomes familiar with the capabilities associated with voice-only input. The vocabulary should support such transitional phases.

Recommendation 2: When possible, include the names of on-screen hyperlinks in the voice command vocabulary. When users wished to directly access links and buttons, they did so by reading the text associated with the link or button. Conversely, links and buttons which contained no text were dealt with via a point-and-click strategy.

Recommendation 3: Whenever possible, structure the voice command vocabulary to take advantage of direct input. For example, ensure that commonly-used functions can be accessed with a minimum number of voice commands.

Recommendation 4: Account for occupation-specific slang (i.e., multiple words and phrases which might apply to the same item). For example, in aircraft maintenance environment, an auxiliary power unit may be referred to as an "auxiliary power unit" or an "A-P-U." T.O.s may be referred to in their entirety, or by their final three or four digits, prefaced by the word "dash." In general, for a given user population, such commonly-used slang or nicknaming should be included as part of the basic voice recognition vocabulary.

## 4.9 Summary

This study represented an initial attempt to develop a vocabulary for a voice recognition system used to interact with ***United States Air Force (USAF)*** maintenance technicians. All subjects were actual maintenance technicians, employed at either Wright-Patterson AFB or the Springfield OANG. These subjects were selected as legitimate end users of such a hands-free electronic maintenance information presentation system. Furthermore, the experimental task involved the use of actual electronic maintenance T.O.s, thus using a source of information with which subjects were familiar.

It is unreasonable to expect any voice command vocabulary to accommodate the strategies of every possible user. However, based upon user feedback, such a system would be acceptable for the presentation and retrieval of electronic maintenance data. The documentation and classification of both user strategies and verbal commands used by maintainers provides attempts to address the issues of a wide range of user tendencies. The resulting word set is a reasonable baseline for the development of a usable and useful vocabulary, and ultimately, an electronic maintenance system which will enhance maintainer performance without compromising safety.

However, before such a system can be implemented, other issues must be studied. Topics such as system hardware design, system usability in the actual maintenance environment, and overall user acceptance can be addressed in a phased methodology. In general, such a test methodology should address the four primary components of a system: ***Software, Hardware, Environment, and Liveware (SHEL)*** (the human user). By addressing each of these primary components in a phased approach, designers can gain a holistic perspective on the usability of such a system from the maintainer's standpoint, as well as the general standpoint of the military aircraft maintenance environment. The final result will be an electronic maintenance system which enhances maintainer performance without compromising safety.

## 5.0 EASE-OF-INTEGRATION INTO USER SYSTEM AND INNOVATIVE APPLICATION OF PROVEN TECHNOLOGY--OBJECTIVES 2 AND 3

These objectives are combined because they both heavily impact the technical design of the software and hardware and encompass the majority of the technical development work. Under these objectives, we determined both the feasibility of the cross-platform design concept and also determined the feasibility of integrating ***commercial off-the-shelf (COTS)*** products (speech software, head-tracker, throat microphone, etc.) to test compatibility of resident speech engines in conjunction with the VHIC software and head-tracking hardware. We also determined the feasibility of the dual-mode speech recognition approach and the usability of watchword signaling as a means of differentiating command/control speech from extraneous speech and/or other noise.

The first objective specifically addressed the critical problem of developing a system that can work in multi-platform environments and allow for hands-free interaction with all of the electronic T.O. modalities (PDF, IETM, HTML, etc.), but without having to wait until the technical data format has been finalized to start our development efforts. As mentioned above, we determined (by literature review and talking to ***Subject Matter Experts (SMEs))*** that the main formats currently being considered for electronic T.O.s are based on PDF (used with Adobe Acrobat Reader) and IETM. However, while no examples currently exist, "web page" based formats (such as HTML) are also being considered for possible future deployment.

The design of the VHIC system allows it to work across all platforms by interacting with any Windows application. This is important because it is highly unlikely that alternative control system designers will have access to the underlying code and layout of the future T.O. software or maintenance logistics systems, but it is likely that these systems will be designed to utilize a Windows platform. Therefore, it is better to design a system that could work with any software designed for ***Personal Computer (PC)*** (Windows) based platforms than to try to design specifically to each application. However, it is also important to allow for application-specific vocabularies and command sets to ease usability. It was with this in mind that we chose to design the VHIC system so that it will transparently replace the mouse and keyboard currently used for desktop applications.

To this end, the VHIC design allows the system to perform simple mouse functions (point, click, double-click, drag and drop, etc.) as well as keyboard functions (text entry, one-key command/control, etc.) and thus allows it to interact with any software without specific set-up requirements. The usefulness of this approach is that the system can interact with maintenance data in all of the formats currently used. We have also added a higher level interaction dialog on top of the basic system to reduce the need for repeating frequent steps. This is equivalent to adding macros to the software interface and does not alter the system's ability to be transparent to the PC and the application software. Many of these macros are built into COTS speech recognizers and we have used these wherever possible. Others were designed and developed specifically for VHIC.

A major advantage to this approach is that it allows development of hands-free wearable systems independent of the development of the application software. This is important because we have little influence over the direction technical data formats will go. We need to be flexible in our implementations, but we cannot afford to wait until the technical data format has been finalized to start our development efforts and it is extremely risky to base development on a limited data platform that may or may not ever be implemented.

Another advantage to this type of approach is that in the rare event of a system failure or an unanticipated environmental condition where the system falters, an emergency, supplemental, wrist-worn keyboard/trackball system could be plugged right in and the user would be able to continue. The swap would be transparent to the computer and the inputs would be identical. If a system were designed that required sophisticated integration with the maintenance software, this transparent swap would not be possible[1].

A third advantage to this approach is that it still allows for the development of new sophisticated multi-modal dialogs and interactions to replace the traditional drop-down menus and button icons. These new graphical interfaces can be optimized for the controller we are using. For example, if desired, a head-slewn cursor could be placed in the top right corner of the display to signify a page change or moved into the lower left corner to bring up a text box. However, the use of these new interaction techniques does not preclude the use of traditional pull-down menus and, furthermore, requires no changes to existing application programs. These techniques or graphics can be added to the system at the device driver level or a resident program can be added that runs concurrent to the application program.

Finally, by developing a transparent system, we help to obtain our fifth objective by greatly increasing the commercial potential of the system, because it could also be used with any non-maintenance-related application software on the market.

## 5.1 Speech Recognition

The speech component of the proposal has many elements. In brief, we have chosen to use a COTS noise dampening throat microphone as the input device. The signal is then parsed through a COTS speech recognition system using "dual-mode" (command and dictation) recognition with watchword signaling. These elements will be elaborated below.

### 5.1.1 Throat Microphones

Throat microphones are small flat microphones worn with a strap on the throat and receive vibration energy which is generated on the skin near the vocal cords. They feature high isolation capability not only from environmental noise, but also from frictional vibrating sound generated by the microphone head. They are light, comfortable, and ideal for use in high-

[1] Note: a wrist-worn keyboard/pointer system would appear to be the most likely supplemental system because of its ease-of-use and familiarity. Chorded keyboards and other non-conventional items would just add to the maintainer's workload. Furthermore, (Thomas, Tyerman and Grimmer, 1997°) report on an experiment comparing a wrist-worn keyboard, a virtual keyboard, and a Kordic keyboard for text entry tasks. They conclude "the forearm keyboard is the best performer for accurate and efficient text entry."

noise situations. We chose the PRYME SPM-500 (Premier Communications, Inc.) because of its cost ($41), availability, styling, and performance. See Section 6.1 for more information about the throat microphone.

### 5.1.2 Speech Recognizer

Aside from the use of noise dampening throat microphones, we believe we can overcome many of the deficits that noise has on speech by using a "dual-mode" speech recognition approach. A dual-mode approach utilizes a limited size robust vocabulary for command and control, but allows for a full utilization of continuous large vocabulary speech recognition for dictation when needed and conditions allow.

There are many COTS speech recognizers available. Choice of platform eliminates some good candidates and the desire for a dictation capability eliminates others (such as Dragon Dictate, Nuance, and Verbex Speech Commander which are command only systems). However, the latest version of Dragon NaturallySpeaking Professional (V4.0 was introduced in the Fall of 1999) was found to have many desirable features as well as superior recognition. However, one problem with the newest NaturallySpeaking speech engine is that it requires computer systems more powerful than current wearable computers. Dragon System's previous version of NaturallySpeaking (3.0) was capable of running on a wearable platform; however, it does not have any built-in mouse functionality and would require a software overlay to mimic mouse inputs. It also has inferior recognition compared to V4.0 and does not allow for the use of macros.

Dragon NaturallySpeaking Professional V4.0 is a very accurate (greater than 99% under ideal conditions) continuous speech recognition system with a total vocabulary of 250,000 terms (of which 160,000 are active at any one time). Inactive words can readily be made active and new words, specialized terms, acronyms, and proper names can be easily added. To aid recognition further, multiple small independent vocabularies could be developed for multiple applications. This allows the recognizer to limit its search saving time and increasing accuracy. One setback of this recognizer is that it is user-dependent and thus requires training. However, the training is minimal and can usually be completed in ten minutes or less per user. One advantage of the training ability of the system is that it allows us to adapt the models to the characteristics of the throat microphone as well as the user. Therefore, performance with the muffled-throat mike signal is not as degraded as it might be for a nonadaptable system. NaturallySpeaking Professional also allows for the development of macros that help automate complex tasks.

VHIC is integrated with NaturallySpeaking using the Dragon NaturallySpeaking ***Software Development Kit (SDK)*** and the MS ***Speech Application Programming Interface (SAPI)***. These two programming interfaces were specifically designed to give programmers the ability to speech-enable Windows applications and can be freely downloaded off of their respective web sites[2]. VHIC could have been developed using only the Dragon NaturallySpeaking SDK; however, MS SAPI was used wherever possible to limit the amount of

[2] Dragon NaturallySpeaking SDK is available at *http://developer.dragonsys.com/* and Microsoft SAPI is available at *http://www.microsoft.com/Products/Speech/SpeechSDK/5 Legal/SpeechSDKEULA.htm*

re-development that would need to be done if VHIC were to be re-written to work with a different speech engine. The advantage of SAPI is that it is a generic programming interface that allows developers to write speech applications that will work with any SAPI-compliant speech engine. The disadvantage of SAPI is that it does not provide the complete access to the Dragon NaturallySpeaking engine that you get with the Dragon NaturallySpeaking SDK. SAPI was designed to be generic and, thus, often does not provide access to functionality that is engine-specific. Therefore, a combination of the two was the best solution for this application.

The first indication that SAPI alone would not be enough occurred when it was realized that "watchword mode" would not always work using only SAPI (see Section 5.1.3 for more about Watchword Signaling). When the NaturallySpeaking built-in commands are turned on (either by VHIC or another speech application such as NaturalWeb), NaturallySpeaking takes control of the microphone and overrides any SAPI microphone control. This problem was solved using the Dragon NaturallySpeaking SDK to put the microphone to sleep after each command during watchword mode. The word "*Computer*" was then added to the list of NaturallySpeaking built-in commands that will wake up the microphone[3].

The majority of VHIC commands are implemented by tapping into functionality provided by the operating system through the MS Windows ***Application Program Interface (API)***. For example, mouse emulation uses this approach. Normally, when the left button on a mouse is clicked, the mouse's driver software detects that the button was clicked and sends a message to the operating system telling it so. The operating system then performs the appropriate action (opens a menu if the click occurred on a menu, closes a window if the click occurred on a close button, etc.). In this case, VHIC takes the place of the mouse driver. When the user says "*click,*" VHIC uses the MS Windows API to send the same message to the operating system informing it that the left mouse button was clicked. The operating system then interprets the message as if it had been sent by the mouse driver.

To speech enable functionality that is specific to an application, a developer must have some sort of access to that functionality. Often, an application is designed to allow developers access to some of this functionality. In the case of NaturallySpeaking, there exists the SDK which can be used to make the application perform certain operations. The applications themselves (e.g., Adobe Acrobat Reader) often have shortcut keys that a user can press in order to get the application to do something. Using the Windows API, these keys can be simulated to perform the operation. NaturallySpeaking itself performs much of its built-in speech operation by simulating keystrokes.

It is in this way that Adobe Acrobat Reader was integrated with VHIC in order to interact with the PDF format T.O.s. Reader was designed to accept shortcut keys and perform specific operations when those shortcut keys are pressed. For example, when viewing a document in Acrobat Reader 4.0, holding down the CTRL key and pressing the minus (-) key causes Acrobat Reader to zoom out on the document; thus when the user says "zoom out," the program simply simulates pressing of the CTRL and minus keys.

[3] The use of the word "*Computer*" as our watchword was completely arbitrary. Any word could be used as explained in Section 5.1.3.

If there is no existing way to get access to the application's functionality, then the developer of the speech application (VHIC in this case) will have to work with the developer of the application in question to have software developed that will provide access to the desired functionality[4].

#### 5.1.2.1 VHIC Command Vocabulary

The complete VHIC command vocabulary itself is listed in Appendix B, "Commands," and defined in the file *Main.grm*. This file can be modified to change the format of current commands without having to recompile the VHIC executable program[5]. Changing the NaturallySpeaking built-in commands can be done by modifying the *global.dvc* file or by using the NaturallySpeaking Edit/New Command Wizard (reference the documentation on "Creating Voice Commands" that is included with Dragon NaturallySpeaking).

#### 5.1.2.2 Context-Specific Vocabularies

To increase recognition accuracy, VHIC utilizes what are called "context-specific vocabularies." This means that a limited (and probably unique) vocabulary set is available to be recognized depending on which window or object the user is interacting. A limited vocabulary set helps improve recognition accuracy and speed by decreasing the available options. For example, the "zoom out" command implemented above would only be available when the Acrobat Reader window has the current focus. Thus, the "zoom out" command would not be understood by the speech engine as a valid command in a window that did not allow zooming.

#### 5.1.2.3 Modes

*"Limited Command Mode."* When limited command mode is active, only the VHIC commands (see Appendix B) are active. The VHIC commands are activated through the SAPI SDK by activating the grammar contained in the file *Main.grm.*

*"Command Mode."* Command mode activates both the VHIC commands and all Dragon NaturallySpeaking built-in commands (see Appendix B). A list of NaturallySpeaking built-in commands can be found in the appendix of the Dragon NaturallySpeaking documentation that comes with Dragon NaturallySpeaking. The built-in commands can be turned on or off using the Dragon NaturallySpeaking SDK. They are turned on or off when VHIC is registered with the NaturallySpeaking speech engine. Therefore, switching between command mode and limited command mode requires VHIC to re-register with the speech engine specifying whether the built-in commands should be on or off.

*"Dictation Mode."* Dictation mode activates the VHIC commands, the built-in NaturallySpeaking commands, and also allows dictation into almost any Windows text control. In other words, wherever you can type, you can talk. Dictation mode is turned on and off using

[4] Note: This was done for both TAPTALK and GCCS COP so ***SYTRONICS*** has a lot of experience working with companies to get this job done.
[5] For help with the syntax of the *Main.grm* file, see the documentation for context-free grammars that is included in the help files of the Dragon NaturallySpeaking SDK and Microsoft SAPI SDK.

the NaturallySpeaking SDK. Dictation can only be turned on and off one window at a time by the SDK. Therefore, if dictation mode is on, every time a new window is brought into focus, dictation is turned on for that window. If dictation mode is off, dictation is deactivated for each newly-focused window.

It is important to note that both the NaturallySpeaking built-in commands and dictation can be turned on by any running application that has been built to work with NaturallySpeaking (such as MS Word, Internet Explorer, Wordperfect, and MS Outlook). Therefore, running such an application at the same time VHIC is running may *cause limited command mode* and *command mode* to function incorrectly.

### 5.1.3 Watchword Signaling

Most current throat microphones and voice-operated systems, utilize either ***push-to-talk (PTT)*** buttons or ***voice-activated (VOX)*** technology to reduce extraneous inputs. Because PTT buttons require some sort of manual activation, they are not feasible for hands-free tasks. On the other hand, VOX systems act by starting the system when a sound is recognized and stopping it when the sound ceases. This has two primary problems--it does not allow for screening of casual speech from command-directed speech and the abrupt onsets and offsets of the VOX systems can cause problems for speech recognition systems. Most VOX technology is used in telephony or other person-to-person applications or for recording speech. These signals are not designed as inputs to a speech engine. However, because it is advantageous to develop a hands-free approach to PTT that is compatible with speech recognition, we implemented the use of a keyword or "watchword" that signals to the computer that the next utterance (word or group of words) is a command (see example in Table 5.1.3-1). The watchword speech system continuously recognizes and parses all inputs, but only acts on legal commands.

**TABLE 5.1.3-1**
**A SAMPLE DIALOG USING THE WATCHWORD "*COMPUTER*"**

| SPEECH INPUT | SUBSEQUENT ACTION |
|---|---|
| "*Computer*, next page" | Screen advances one page |
| "*Computer*, scroll page up" | Graphic scrolls up one page |
| "scroll page down" | <no change> |
| "*Computer*, quit watchword mode" | <*WATCHWORD* mode turned off> |
| "scroll page down" | Graphic scrolls down one page |
| "next page " | Screen advances one page |
| "*Computer*, next page" | Screen advances one page |
| "take note" | Note Box appears on the screen; system automatically switches to *DICTATION* mode |
| "The quick brown fox jumps over the lazy dog." | "The quick brown fox…" is written into text box |
| "save note" | System saves note to next file (see Appendix B command description) and goes back to *COMMAND* mode |
| "Watchword mode" | <starts *WATCHWORD* mode> |
| "next page" | <no change> |
| "*Computer*, next page" | Screen advances one page |
| {etc.} | {etc.} |

Users can toggle the use of a watchword off (for times when speech is only directed to the computer) and on (when extraneous speech may be erroneously recognized). Also, certain commands will automatically toggle the mode off temporarily (for example, when using the "*Find*" voice command in Adobe Acrobat Reader; see Section 5.4 below).

Watchword mode was implemented by using the ability of the NaturallySpeaking SDK to put the microphone to sleep. Upon the command "*watchword mode,*" the microphone is put to sleep. The words "*computer*" and "*wake up*" were added to the NaturallySpeaking built-in commands in the *global.dvc* file and performs the operation of waking up the microphone. Both commands will wake up the system, however, using "*computer*" will cause the system to go back to sleep after the next utterance while using "*wake up*" will keep the microphone awake until it is actively put back to sleep again.

The actual watchword chosen is arbitrary and can even be implemented as to be user-changeable (although this has not been done for the concept demo). It is important only to have something simple, memorable, and yet uncommon so as not to come up in normal everyday speech. The use of the word "*computer*" here would probably not be satisfactory in most implementations as it is a fairly common word in technical environments. Any word can be used. Even nonsense words invented by the user could be used because any word can be added to the NaturallySpeaking vocabulary and easily trained.

In *limited command mode*, the NaturallySpeaking built-in commands are turned off. This, unfortunately, also turns off the watchword command "*computer.*" This invalidates the above implementation of *watchword mode*, causing the need for a special case implementation of *watchword mode* during *limited command mode*. During *limited command mode*, the command "*watchword mode*" disables the active grammar and activates a grammar containing the sleep command "*computer.*" When the user says the word "*computer,*" the previously-active grammar is re-activated and the sleep grammar is deactivated. Then, upon the next utterance, the active grammar is again deactivated and the sleep grammar becomes active once again. It is important to note that this implementation will not work if another running application is using NaturallySpeaking and has loaded the built-in commands; because when the VHIC sleep grammar is active, the NaturallySpeaking built-in commands will still be active and *watchword mode* will, thus, not function as designed. In this case, the built-in commands will continue to be active, therefore, *limited command mode* serves no purpose and *command mode* should be used in its place.

## 5.2 Head Control

For one type of system input--cursor pointing--eye- and head-based controllers can harness the tendency of humans to naturally orient towards objects they wish to control. However, speed and accuracy improvements in eye-tracking technologies are required before eye-based control can be realized for present HMD systems. Head-based controllers provide a solution for practical hands-free cursor positioning. Small, lightweight, transmitter-less head-trackers that utilize inertial sensors can be easily and unobtrusively mounted onto current HMDs and provide precise smooth cursor control using only small head movements. Unlike magnetic, optical, and sonic trackers, inertial trackers do not require any elements to be mounted in the environment away from the user. Furthermore, cursor pointing tasks do not require full six degree-of-freedom input--simple two-axis pitch and yaw is all that is required to move a cursor horizontally and vertically across a display. This makes inertial trackers an ideal low-cost solution to hands-free pointing for wearable computers. For our system, we have chosen a MicroGyro 100 Sourceless Tracker (Gyration, Inc.)--a low-power, sourceless tracker module utilizing a two-axis inertial gyroscope system to determine pitch and yaw information (see Figure 5.2.1). When packaged as the GyroPoint Pro mouse (priced at $129 OTS), its output is identical to that of a MS Mouse. The small (2.2 x 2.2 x 2 cm) cube can be removed from the assembly and remotely mounted on a HMD or GMD requiring only a cable connection to the electronics board. The cube, board, and cabling can be repackaged to be better integrated into a wearable system. The MicroGyro system requires only a serial or PS2 port for connection to the computer.

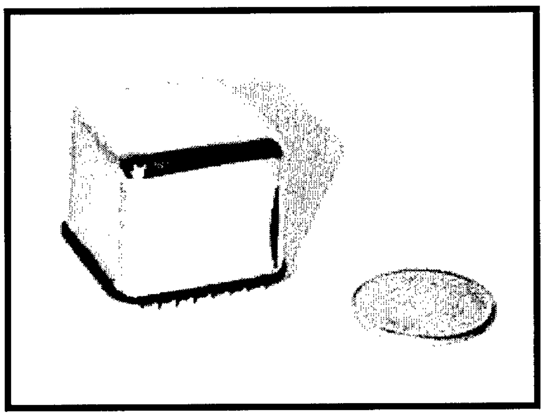


**Figure 5.2-1. Gyration 100 Cube**

The output of the head-tracker behaves exactly like a standard mouse output utilizing relative positioning[6]. In relative positioning, the cursor moves in the same direction as the head movement, but the extent of the relative cursor motion is dependent on a specific gain applied to the input. The cursor is further constrained by the edge of the display allowing users to re- "calibrate" the system by using the edge of the display. This style of cursor positioning was used in the ***SYTRONICS***/UDRI field study mentioned above and produces satisfactory results. We have further added verbal commands to re-center the cursor and verbally adjust the speed (or gain) of the cursor motion. The command "*center mouse*" causes the cursor to jump from wherever it is to the center of the display helping to reduce head movements. The commands "*move mouse slower*" and "*move mouse faster*" change the gain by small increments up or down

[6] We did not design the system to use "Joystick"-style control which uses a different approach. Here, there is a neutral head position (straight ahead) that results in no cursor movement. When the user wishes to move out of the neutral area, he moves his head in the required direction and holds his head in that direction until the cursor is positioned correctly. The user then returns his head position to the dead area to stop the cursor movement. Speed and acceleration of the cursor can be made relative to the extent of the head movement in the given direction. However, the use of such a device is problematic for a maintainer who is constantly moving his head around while completing tasks. While the ability to verbally turn the pointing system on and off would reduce problems, this style of interaction may only be useful for maintenance applications that require scrolling through large graphics or schematics (similar to using a scrolling wheel on a mouse), although this can already be done using the verbal scrolling commands (see Appendix B).

allowing for user preference. It has been shown (Lin, Radwin, and Vanderheiden, 1992[n]) that gains on the order of 0.3-0.6 (one degree of head motion = 0.6 degrees of cursor movement) produce the minimum movement time and lowest ***Root Mean Squared (RMS)*** cursor deviation; however, we have given the user the ability to change gains over the entire range (gains near zero to over 5.0). Furthermore, the command "*slowest mouse speed*" automatically reduces the gain to the minimum available allowing for very small adjustments in cursor placement. Finally, saying "*default mouse speed*" returns the gain to the default MS Windows setting.

## 5.3 VHIC Commands--Interacting with PDFs using Adobe Acrobat Reader

VHIC has been optimized for the concept demonstration to work with Adobe Acrobat Reader to interact with PDF files. It is also capable of interacting with IETM in a simple mouse emulation mode and with HTML files using MS Internet Explorer and built-in NaturallySpeaking commands. However, we chose to focus on the PDF format for the concept demo because of the availability of PDF T.O.s.

### 5.3.1 Mouse Emulation

VHIC baseline function will work with all Windows applications which enables complete mouse emulation of pointing and clicking including dragging, dropping, etc. The VHIC mouse emulation commands are listed in Table 5.3.1-1.

**TABLE 5.3.1-1**
**VHIC MOUSE EMULATION COMMANDS**

**Selection Commands**

| SPEECH | RESULT |
|---|---|
| "Double click" | Double clicks the left mouse button at the current cursor position. |
| "Click"<br>"Left click"<br>"Select"<br>"Enter"<br>"Open" | Single clicks the left mouse button at the current cursor position. These commands can be said while the microphone is asleep or while VHIC is in watchword mode. |
| "Drag"<br>"Left button down" | Presses and holds the left mouse button at the current cursor position. |
| "Drop"<br>"Left button up" | Releases the left mouse button. |
| "Right click" | Single clicks the right mouse button at the current cursor position. |
| "Middle click" | Single clicks the middle mouse button at the current cursor position. |

**Pointing Commands**

| SPEECH | RESULTS |
|---|---|
| "Move mouse slower" | Decreases the cursor speed relative to the amount of movement by the mouse or head-tracker (not available using Win95). |
| "Move mouse faster" | Increases the cursor speed relative to the amount of movement by the mouse or head-tracker (not available using Win95). |
| "Reset mouse speed" | Resets the cursor speed to its default value (not available using Win95). |
| "Slowest mouse speed" | Decreases the cursor speed to its minimum (not available using Win95). |

| SPEECH | RESULTS |
|---|---|
| "Mouse left" | Starts the cursor moving left. |
| "Mouse right" | Starts the cursor moving right. |
| "Mouse up" | Starts the cursor moving up. |
| "Mouse down" | Starts the cursor moving down. |
| "Faster" | Moves the cursor faster. |
| "Slower" | Moves the cursor slower. |
| "Stop mouse" | Stops the cursor movement. |
| "Center mouse" | Moves the cursor to the middle of the screen. |
| "Go left" | Moves the cursor left a small distance. |
| "Go right" | Moves the cursor right a small distance. |
| "Go up" | Moves the cursor up a small distance. |
| "Go down" | Moves the cursor down a small distance. |

These commands are designed to duplicate, using speech and head pointing, exactly what the user would do with a mouse--with a few cursor pointing commands added to accent the use of the head-tracker.

### 5.3.2 Scrolling, Zooming, and Paging

As a next level of interaction, we have added the scrolling, zooming, and paging commands identified in the Objective 1 study (Table 5.3.2-1). We used the most popular command words identified in the study whenever possible, although sometimes extra options are included. For example, while the subjects preferred to use the term "Scroll down" to scroll down one line, we added the term "Scroll line down" to do the same thing. Similarly, you can scroll down the extent of the window by using "Scroll page down" or simply "Page down." If desired, all of the commands can be easily changed or new ones added.

**TABLE 5.3.2-1**
**VHIC SCROLLING, ZOOMING & PAGING COMMANDS**

**Scrolling Commands**

| SPEECH | RESULT |
|---|---|
| "Scroll up"<br>"Scroll line up" | Scrolls up a line. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll down"<br>"Scroll line down" | Scrolls down a line. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Page up"<br>"Scroll page up" | Scrolls up a page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Page down"<br>"Scroll page down" | Scrolls down a page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll left" | Scrolls left by one unit. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll right" | Scrolls right by one unit. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |

| SPEECH | RESULT |
|---|---|
| "Scroll page right"<br>"page right" | Scrolls right one page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll page left"<br>"page left" | Scrolls left one page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling left" | Starts automatic scrolling to the left. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling right" | Starts automatic scrolling to the right. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling up" | Starts automatic scrolling up. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling down" | Starts automatic scrolling down. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Stop scrolling" | Stops automatic scrolling. |
| "Speed up" | Makes automatic scrolling scroll faster. |
| "Slow down" | Makes automatic scrolling scroll slower. |

## Zooming Commands

| SPEECH | RESULT |
|---|---|
| "Zoom in"<br>"Half scale" | Zooms in at the current cursor position. This command is only valid if Acrobat Reader is open and has focus. |
| "Zoom out"<br>"Double scale" | Zooms out from the current cursor position. This command is only valid if Acrobat Reader is open and has focus. |

## Paging Commands (*Acrobat Reader only*)

| SPEECH | RESULT |
|---|---|
| "Next page" | Displays the next page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "Previous page" | Displays the previous page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "First page" | Displays the first page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "Last page" | Displays the last page of a document. This command is only valid if Acrobat Reader is open and has focus. |

The scrolling commands should work within many Windows applications, but not all. This is due to the fact that scrolling can be implemented in many different ways. Two of these implementations allow for it to be easily determined that the window is scrollable and allow the ability to simulate scrolling though the MS Windows API. For example, some applications send a message to the window telling it to scroll when the scroll bar is moved. Many scrollable windows allow us to "easily" simulate this message through the Windows API. Only windows that were implemented in one of the two ways that use the Windows API have been made scrollable through VHIC.

The zooming and paging commands, however, work only in Adobe Acrobat. Zooming and paging are not capabilities provided by MS Windows. Zooming, like "Next Page," "Previous Page," etc., is a functionality that is specific to Acrobat Reader. While another application may have the ability to zoom, its implementation probably has nothing in common with the zooming capability of Acrobat Reader. Zooming was speech-enabled through the simulation of key strokes that are unique to Acrobat Reader. Another application that can zoom has not necessarily been implemented to accept those same key strokes.

### 5.3.3 Searching in Adobe Acrobat Reader

We have also tapped into the functionality of Adobe Acrobat Reader to allow a voice-operated search capability. Acrobat Reader has a "Find" feature (accessed by the binocular icon on the desktop) to bring up a dialog box in order to search for a word or phrase (see Figure 5.3.3-1). We have speech-enabled this feature using the VHIC command "*Find.*" This will now bring up the dialog box and allow voice interaction. When the box is brought up, the system automatically switches to dictation mode and activates the "Find What:" text box. The next recognized utterance will then be placed into the text box unless that utterance is another VHIC command. In that case, the VHIC command would have priority over the dictation fill-in. To start a search, the user simply says "*go*" or "*start search.*" Before starting a search, the user can also toggle on or off the various options by simply uttering them. For example, saying "*match case*" will toggle the Match Case option, "*find backwards*" toggles the "Find Backwards" option, etc. The VHIC Find Commands are listed in Table 5.3.3-1.

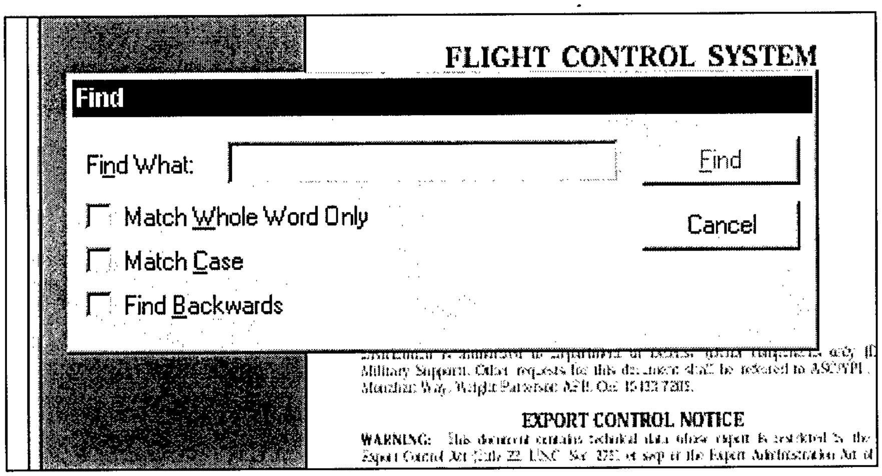



**Figure 5.3.3-1. Adobe "Find" Dialog Box**

(REMAINDER OF PAGE INTENTIONALLY LEFT BLANK.)

**TABLE 5.3.3-1**
**VHIC FIND COMMANDS**

***(Acrobat Reader only)***
These commands are only valid if the Acrobat Reader find dialog box is open and has focus.

| SPEECH | RESULT |
|---|---|
| "Find"<br>"Find word" | Opens the find dialog box. |
| "Find again"<br>"Find next" | Resumes searching for more occurrences of the phrase from the previous search. |
| "Go"<br>"Start search" | Begins searching for the words entered into the find dialog box. |
| "Cancel" | Cancels the search and closes the dialog box. |
| "Match whole word only" | Toggles the "Match Whole Word Only" check box of the find dialog box. |
| "Match case" | Toggles the "Match Case" check box of the find dialog box. |
| "Find backwards" | Toggles the "Find Backwards" check box of the find dialog box. |
| "Find what" | Gives keyboard focus to the "Find What" field of the find dialog box. |
| "Delete line"<br>"Delete that" | Deletes the entry in the find dialog box. |

After a search has been implemented using the Find feature, it can be repeated by simply saying "*find again*" or "*find next.*" A new search can be started by saying "*find*" to bring up the dialog box again.

When using the Find feature in *watchword mode*, *watchword mode* is automatically turned off when the Find dialog box is open. This way, the user can interact with the box without having to preface each utterance with the watchword. Once the search has started (after the user utters "*go*" or "*start search*"), then *watchword mode* is re-implemented so that in order to do another search, the user would have to say "*computer… find again,*" etc.

We have also added the commands "*delete line*" and "*delete that*" which allow the user to delete the last utterance from the text box. For example, if a user opens the Find dialog and then utters "*flight control*" to search for any instances of the phrase "flight control," the recognizer may misinterpret the utterance as, say "light control." If so, the user can say "*delete line*" or "*delete that*" to clear the box. These are VHIC commands, but it should be noted that whenever dictation mode is turned on (as it is automatically when using the Find dialog box), the user has access to all of the NaturallySpeaking editing commands. If this occurs, the user can simply say "*Scratch that*" to erase the last utterance and try again. A few useful commands are listed below in Table 5.3.3-2. Other NaturallySpeaking editing commands and syntax can be referenced in Appendix B of the *Dragon NaturallySpeaking User's Guide.*

**TABLE 5.3.3-2**
**NATURALLYSPEAKING EDITING COMMANDS**

| SPEECH | RESULT |
|---|---|
| "Scratch that" | Erases the last utterance in a text (dictation) box |
| "Backspace"<br>"Backspace <#>" | Deletes the last character in a text box or the last # of characters. Example -- "Backspace five" deletes the last five characters. |

### 5.3.4 Navigation

While VHIC has a head-tracker/mouse pointer with which to navigate around the screen, NaturallySpeaking has many built-in navigation features that can aid usability and tie in to the Objective 1 study results (see Table 5.3.4-1). For example, to get to the FIT VISIBLE option under the View menu (see Figure 5.3.4-1) using the mouse, you would first click on VIEW, then click on FIT VISIBLE. NaturallySpeaking gives you the option of saying "*Click view,*" then "*Fit visible*" to perform the task without pointing. All menu items can be accessed this way by saying "*Click*" followed immediately by one of the menu item names such as FILE, EDIT, HELP, etc. Once the drop down menu appears, any of the options can be accessed by simply saying the word(s). For example, to save a document you would simply say, "*Click file*" then "*Save.*" It is important to say the menu item immediately after the word click, because if you pause after saying the word "*Click,*" then the system will interpret it as a left mouse click.

**TABLE 5.3.4-1**
**NATURALLYSPEAKING NAVIGATION/KEYBOARD COMMANDS**

| SPEECH | RESULT |
|---|---|
| "Click <button or menu name>" | Activates any button or menu item in the active window. Examples – "Click OK," "Click Cancel," "Click File," etc. |
| "Press <key name>" | Duplicates activation of a keyboard key press. Examples -- "Press Tab," "Press h," "Press Shift F7," etc. |
| "Cancel" | Closes a menu or undoes a previous command. |

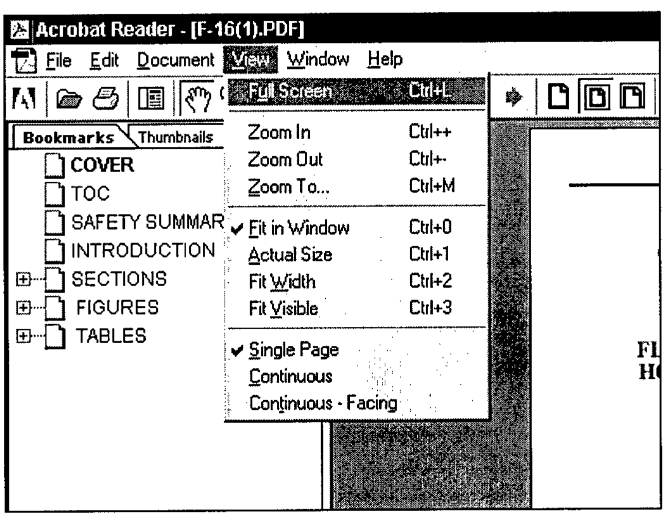



**Figure 5.3.4-1. Sample Drop Down Menu**

Another useful NaturallySpeaking feature is the ability to mimic keystrokes by simply uttering the key name after saying the word "*Press.*" This works for single keystroke ("*Press g,* "*Press Caps Lock*") or for multi-key combinations ("*Press shift F7,*" "*Press Control Alt Enter*").

### 5.3.5 Note Taking and Form Filling

Finally, to demonstrate VHIC's ability to do command/control and dictation, we have added two pop-up features that are not normally part of Adobe Acrobat Reader.

- Note Box: The first is a note taking dictation box (Table 5.3.5-1). When the user says "*take note,*" an empty text box will pop up and *dictation mode* will be turned on. The user can then dictate a note into the box--editing and formatting it as with any other NaturallySpeaking document. When the note is complete, the user says "*save note*" to save the note to disk or "*quit note*" to quit the note box without saving. If the note is saved, it is stored in the directory \VHIC\Notes in a file called *<date>.txt* where *<date>* is the current date (as stored in the CPU memory). Subsequent notes on the same date will be appended to the file. As when using the "Find" box, *watchword mode* is automatically turned off when you bring up the note box and is not implemented again until the box is closed.

**TABLE 5.3.5-1**
**NATURALLYSPEAKING NOTE BOX COMMANDS**

| SPEECH | RESULT |
|---|---|
| "Take note" | Opens a note box in which to dictate. |
| "Save note" | Saves the contents of the note box to a file and closes the note box. The file is named after the current date (ex. 091500.txt for notes saved on Sept. 15, 2000) in the directory \VHIC\Notes. Subsequent notes on the same date will be appended to the file. |
| "Quit note" | Closes the note box without saving the note. Any text written to the note box is lost. |

- Form Fill-In: We have also added a sample Part Order Form (Figure 5.3.5-2) to show how VHIC might be used to fill-in standard forms. Any time VHIC is running, you can bring up the form by saying "*Order Part.*" The various parts are filled in by saying the descriptor followed by the information. For example, to fill in the first box, the user would say "*Part Number 613-27.*" The description (Lt. Exhaust Nozzle) will automatically be written in when a valid part number is entered. Currently 613-27, 111-11, and 123-45 are the only valid part numbers for demo purposes, although any number can be entered (see Table 5.3.5-2 or Appendix B for further information). The other fields can then be filled in and the order form can be accepted or canceled.

(REMAINDER OF PAGE INTENTIONALLY LEFT BLANK.)

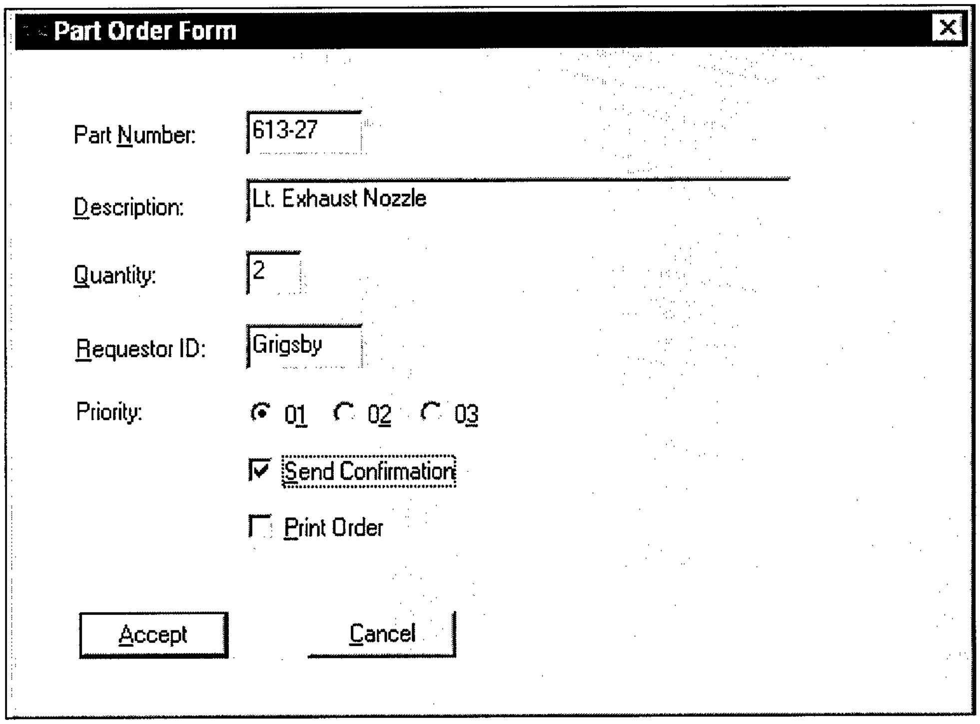



**Figure 5.3.5-1. Sample Part Order Form**

**TABLE 5.3.5-2**
**PART ORDER FORM COMMANDS**

| SPEECH | RESULT |
|---|---|
| "Order part" | Starts the application OrderPart.exe bringing up the Part Order Form window. |
| "Part number <Part_Number>"<br><br>Ex. "Part number six one three dash two seven." | Fills in the Part Number field with the specified <Part_Number>. |
| "Quantity <Quantity>"<br><br>Ex. "Quantity four" or "Quantity one three". | Fills in the Quantity field with the specified <Quantity>. |
| "Requestor I D <Requestor>"<br>"Requestor <Requestor>"<br><br>Ex. "Requestor I D Grigsby" | Fills in the Requestor ID field with the specified <Requestor>. |
| "Priority <Priority_Number>"<br><br>Ex. "Priority one" | Checks the radio button specified by <Priority_Number>. |
| "Send confirmation" | Toggles the Send Confirmation check box. |
| "Print order" | Toggles the Print Order check box. |
| "Accept" | Clicks the Accept button. |
| "Cancel" | Clicks the Cancel button. |

## 5.4 System Requirements

The system was optimized for Windows 98; however, VHIC can be successfully executed on any Windows platform where NaturallySpeaking can successfully run (Windows 95, 98, 2000, and NT). However, many VHIC commands tap into functionality provided by the operating system. Since not all of the same functionality exists across all Windows operating systems, current and future VHIC commands may not be available across all operating systems. At this time, the only commands that do not work across all of the above operating systems are the commands to change the speed of the mouse/head-tracker. These commands are not available on Windows 95, but are available on the Win98/NT/2000.

For the concept demonstration at the conclusion of the Phase I effort, we will not be testing on a wearable system. The software we chose to use has system requirements that are currently beyond the scope of COTS wearable systems. However, because of the numerous advantages of using NaturallySpeaking and the robustness of the models, we felt that we should use the higher-powered software to show the capabilities of our system, and we anticipate the development of wearables with suitable power in the near future. The ability to demonstrate the VHIC concept and develop the software took precedence over the desirable, but less important ability to demonstrate the wearable aspect of the system. The next-generation wearables, along with suitable packaging of the VHIC hardware components, will lead to a compact, robust, and completely wearable system.

### 5.4.1 System as Demonstrated

- Hardware:
  - Dell Inspiron 3700
    - CPU: Pentium III 400 MHz
    - Disk Space: 8 GB
    - Operating System: Windows 2000

- OTS Software:
  - Dragon NaturallySpeaking Professional 4.0
  - Dragon NaturallySpeaking SDK
  - Microsoft Speech API
  - Adobe Acrobat Reader (V4.0)[7]

- Other files needed to run VHIC are:
  - *VHIC.exe*--VHIC executable program
  - *Main.grm*--Grammar file
  - *computer_ready.wav*--Wav file played during watchword mode
  - *global.dvc*--NaturallySpeaking file with modifications to the built-in commands

[7] Note: Features and command keystroke mappings are different between Version 3.0 and 4.0 of Acrobat Reader. The VHIC software has been configured to work with V4.0. Adobe Acrobat Reader 4.0 is available as a free download from Adobe at *www.adobe.com/products/acrobat/readermain.html.*

#### 5.4.2 Minimum System Requirements

- IBM® compatible PC with 300 MHz Intel Pentium processor with MMX, or equivalent; or Windows®95/98 or Windows®NT 4.0 (with SP-3 or greater)
- 48 MBytes RAM for Windows95/98 (64 MBytes is recommended); 64 MBytes for WindowsNT
- 200MBytes free hard-disk space; additional 20 to 50 MBytes required per user
- CD-ROM drive required for installation
- Creative Labs® Sound Blaster® 16 or equivalent sound board supporting 16-bit recording

### 5.5 Summary

VHIC, using NaturallySpeaking and the GyroPoint Mouse, provides a good solution for achieving our objectives. However, a few caveats merit brief mention. First, our goal was to produce a system that was truly transparent to any application. This was accomplished using our mouse emulation features. However, as our study showed, users want more than just the ability to interact with the applications using mouse emulation alone. One benefit of speech as a control mechanism is its ability to simplify multi-step processes. Instead of having to click on FILE, then SAVE AS, and then type a filename, an elegant implementation of speech allows the user to simply say "Save As document1" and it is done. While NaturallySpeaking gives us this power for many commands, others are too application-dependent. Even something as simple as scrolling can be implemented in many ways and it is not feasible to design a system that will speech-enable them all. Mouse emulation gives us the ability to interact with these in a hands-free manner, but it may prove too inelegant for many users. We have shown that for some applications, like Adobe Acrobat Reader, given the development time and effort, elegant higher-level macros can be easily implemented. But others, like the IETM, have to rely on simple mouse emulation alone (although with text fill-in capability). With current operating systems and standards the way they are, these will remain application-specific.

Finally, initial testing of Dragon NaturallySpeaking shows it to be suitable for our requirements and will be able to perform satisfactorily as both a command/control recognizer and a continuous speech system for inclusion in the concept demonstration. However, as part of a Phase II effort, it may be advantageous to look into developing our own speech engines using a Hidden Markov Model development system such as Entropic Inc.'s ***Hidden Markov Toolkit (HTK)*** or the ***Institute for Signal and Information Processing's (ISIP) Automated Speech Recognition (ASR)*** (which is a public domain system). While costlier because of increased development time, building our own engines would allow us to construct an extremely robust, speaker-independent, small vocabulary, command/control system that has been directly trained using data files recorded by the throat microphone under the correct noise conditions. This would dramatically improve recognition, but would also limit the applicability to scenarios similar to those used during training.

## 6.0 SYSTEMATIC TESTING OF SYSTEM ELEMENTS--OBJECTIVE 4

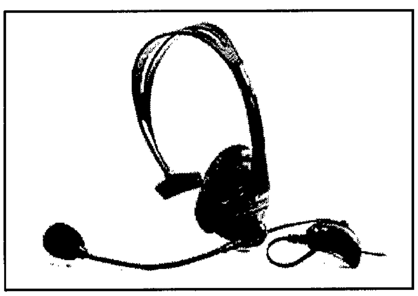
We sought to test all elements of the system by collecting empirical data on individual task parts, comparing and contrasting elements of the system, and testing the interaction of multiple elements in a controlled environment. However, due to time constraints on developing a working concept demonstration, we were unable to provide the level of testing we desired. However, in Phase II, this objective forms the crux of our approach because it provides the platform on which we continually test the feasibility and usability of all aspects of our design. Through continual testing with experts and users alike, we specifically address the critical problem issue of developing a needed system and maintaining user acceptance.

During our evaluation, we were able to significantly test the speech recognizer and develop the functionality we desired. We have also used the head-tracker in previous research (McMillan, Calhoun, Masquelier, Grigsby, Quill, Kancler, Nemeth and Revels, 1999[f]). Below, we relate the microphone comparison test between the throat mike and a baseline boom mike.

### 6.1 Microphone Comparison

As part of our system testing, we conducted a comparison of the chosen throat microphone--the PRYME SPM-501 with a standard OTS boom mike designed to be used with speech recognition software. The two microphones are:

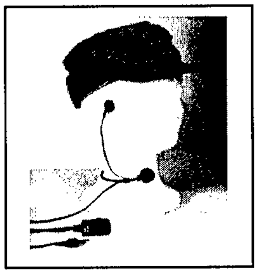
Boom Microphone: Labtec-8450



Specifications:
- 30 mm dynamic mylar diaphragm driver
- output: -67 dBV/microbar, -47 dBV/Pascal +/- 4 dB
- Mic power source voltage: 1.5 VDC
- Impedence: 2000 ohms
- Frequency Response: 100-16,000 Hz

The LVA-8450 ClearVoice™ is a good quality $40 headset/boom microphone designed to provide accurate audio input for PC speech recognition and all other PC multimedia applications. Designed specifically for use with voice command, internet communications, and speech recognition applications--one of the most demanding environments for a computer microphone, by using "NCAT2," which stands for "Noise Canceling and Amplification Technology, Version 2." The microphone has been optimized for speech frequencies, produces a good, strong signal, and its noise-canceling characteristics mean that you should be able to dictate intelligibly to your PC in an office environment with few errors.

Throat Microphone: PRYME SPM-501



The PRYME Radio Products SPM-500 series throat microphone (PREMIER Communications) rests comfortably against the user's throat, picking up audio directly from vocal cord vibrations. This provides outstanding background noise suppression with reasonable clarity. The microphone has a built-in, in-the-ear speaker and an in-line PTT button that can be clipped to the lapel or belt. The PTT button can be bypassed so that the microphone is continuously on.

The Labtec microphone plugs directly into the computer's sound card; however, the PRYME throat mike comes with a connector designed for hand-held radios. This connector features two pins, one standard stereo 1/8" mini phone plug for microphone input and one mono 3/32" submini phone plug for speaker output. An inexpensive 1/8" stereo headphone extension cord was used to plug the throat microphone into the computer with no apparent loss-of-signal.

Sound quality of both microphones was checked using Dragon's Advanced Audio Set-up feature[8]. A comparison of the relative frequency response curves for the PRYME and Labtec mikes are shown in Figure 6.1-1. The upper green bars show the signal response and the lower red bars show the noise response. Both microphones had a signal-to-noise rating of 24 and were rated as having "Average" sound quality with a "High" volume level. Note, however, that the throat microphone (Figure 6.1-1, left) has a reduced frequency response for mid and high frequencies (although this is partially offset by the reduced noise) as would be expected due to its placement on the throat and the fact that it has to rely on vibrations transmitted through the skin and muscles. Because of its frequency response characteristics, this tissue effectively acts as a low pass filter thus reducing speech quality and giving the speech a muffled quality.

(REMAINDER OF PAGE INTENTIONALLY LEFT BLANK.)

[8] This feature is brought up by running NaturallySpeaking's standard *Audio Set-up Wizard* and hitting ALT-1 at the Introductory page.

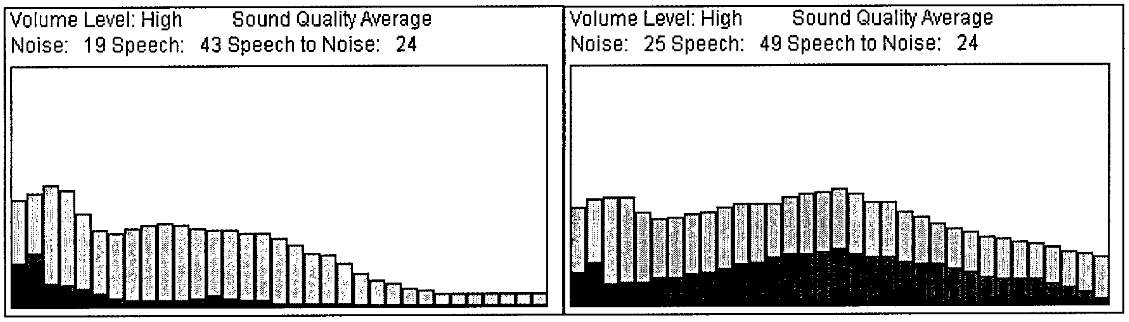



**Figure 6.1-1. Frequency Repsonse**

**Relative frequency response curves of the signal (green) and noise (red) for the PRYME SPM-501 throat mike (left) and Labtec LVA 8450 (right) as measured through the system using Dragon's advanced Audio Set-up Wizard.**

Another problem with using the throat microphone is due to its placement away from the main speech outlet (the mouth) and its reliance on secondary transduction. Because the microphone only picks up skin vibrations relayed from the laryngeal vibrations during speech, certain speech sounds that do not require the larynx to vibrate (the "voiceless" phonemes) are missed entirely. Figure 6.1-2 shows comparison waveforms for the utterance "*This is a speech sample*" as recorded with the Labtec boom mike (upper curve) and the PRYME throat mike (lower curve). Note the obvious loss of signal for the voiceless allophones "*s*" (an alveolar fricative) and "*ch*" (a palatal affricate) as highlighted by the red boxes and arrows. This shows the loss that will necessarily occur for all of the voiceless fricatives ("*f*," "*th,*" "*sh,*" and "*h*") that do not require laryngeal vibration. This loss of signal does not appear to be as prevalent for the voiceless stops ("*p,*" "*t,*" and "*k*") because the percussive signal is still transmitted to the throat. While speech recognition performance using NaturallySpeaking does not appear to be significantly effected by this loss, this effect should be kept in mind as recognition errors occur most notably for ingular/plural discrepancies where the added "s" may be missed (e.g., "flight control" vs. "flight controls," "valve" vs. valves," etc.).

(REMAINDER OF PAGE INTENTIONALLY LEFT BLANK.)

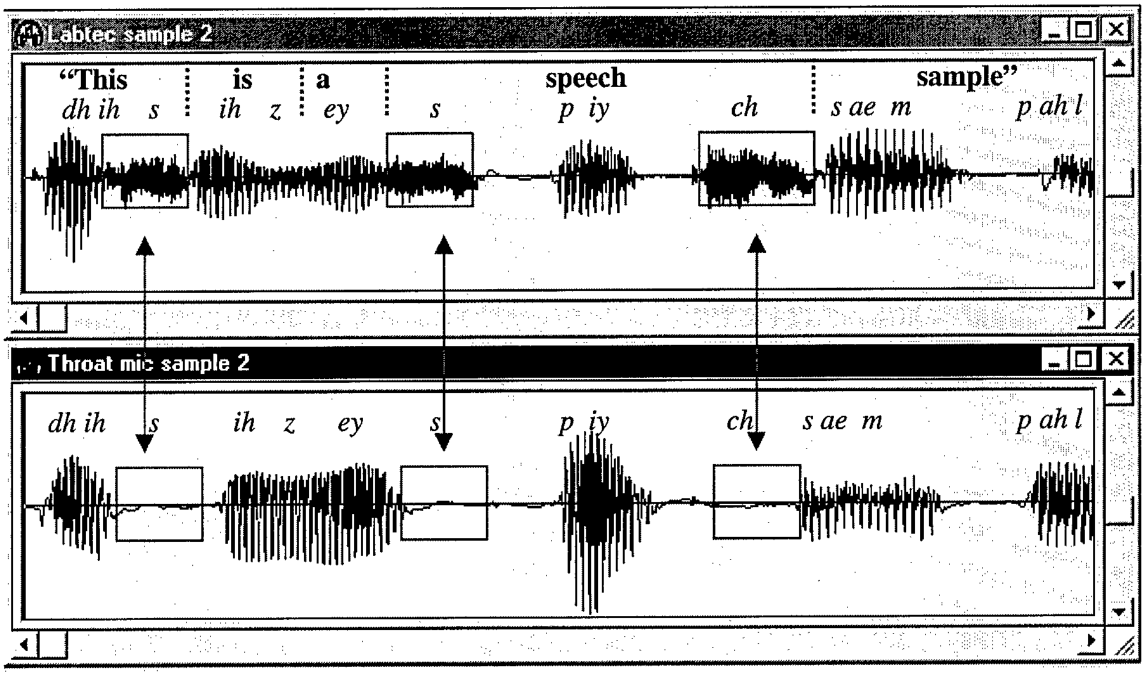



**Figure 6.1-2. Signal Comparison**

**Signal Comparison of the Labtec 8450 boom mike and PRYME SPM-501 throat mike for the utterance "*This is a speech sample.*" Note the loss of signal when using the throat mike for the voiceless phones "*s*" and "*ch.*" See text for details.**

## 7.0 DETERMINATION OF COMMERCIALIZATION POTENTIAL--OBJECTIVE 5

Under this objective, we interacted with potential users and commercial partners to determine the commercial product potential of VHIC. We used information from these sources to ensure our Phase I-level designs and tests were focused upon commercial markets as well as satisfying USAF needs. As a result of achieving this objective, we produced a preliminary feasibility assessment to form a sound foundation for subsequent commercialization. In the course of developing the feasibility assessment, numerous military and commercial applications for VHIC were identified. Some of the identified applications have markets waiting for them right now (requirements pull) while others are breakthrough applications (technology push) and would require educating the potential customers to the benefits of using VHIC. Listed below is a representative list of potential military and commercial applications:

- Maintenance wearable systems
- Command and control (datawall) environments
- 21st century soldier
- Medical personnel in the field

- Sace, HAZMAT, chemical/biological (any application where data access is required while wearing protective gear in a hostile environment)
- Computer access for the disabled
- Gaming control
- Immersive virtual environments
- Emergency Room personnel
- Manufacturing, process and quality control, inventory

In summary of our Phase I Objectives and the results of achieving them, we have demonstrated the basic feasibility of the VHIC concept--to provide a hands-free wearable computer control system for maintenance applications. The Phase I project was specifically structured to be narrow in scope, but of sufficient depth to set the foundation and provide a roadmap for a full-scope development and validation effort in Phase II. To set the stage for this Phase II effort, we next describe our specific definition of the Phase II problem, which is based upon what we accomplished and learned in Phase I.

## 8.0 FEASIBILITY ASSESSMENT AND CONCLUSIONS

This development effort and study represented an initial attempt to develop a wearable control system and vocabulary for a voice recognition system used to interact with USAF maintenance technicians. All subjects used in the study were actual maintenance technicians, employed at either WPAFB or the Springfield OANG. These subjects were selected as legitimate end users of such a hands-free electronic maintenance information presentation system. Furthermore, the experimental task involved the use of actual electronic maintenance T.O.s, thus, using a source of information with which subjects were familiar.

It is unreasonable to expect any voice command control system to accommodate the strategies of every possible user. However, based upon user feedback, such a system would be acceptable for the presentation and retrieval of electronic maintenance data. The documentation and classification of both user strategies and verbal commands used by maintainers provide attempts to address the issues of a wide range of user tendencies. The resulting word set is a reasonable baseline for the development of a usable and useful vocabulary; and ultimately, an electronic maintenance system which will enhance maintainer performance without compromising safety.

However, before such a system can be implemented, other issues must be studied. Topics such as system hardware design, system usability in the actual maintenance environment, and overall user acceptance can be addressed in a phased methodology. In general, such a test methodology should address the four primary components of a system--Software, Hardware, Environment, and Liveware--the human user (SHEL). By addressing each of these primary components in a phased approach, designers can gain a holistic perspective on the usability of such a system from the maintainer's standpoint, as well as the general standpoint of the military aircraft maintenance environment. The final result will be electronic maintenance system which enhances maintainer performance without compromising safety.

VHIC, using NaturallySpeaking and the GyroPoint Mouse provides a good solution for achieving our objectives. However, a few caveats merit brief mention. First, our goal was to produce a system that was truly transparent to any application. This was accomplished using our mouse emulation features. However, as our study showed, users want more than just the ability to interact with the applications using mouse emulation alone. One benefit of speech as a control mechanism is its ability to simplify multi-step processes. Instead of having to click on FILE, then SAVE AS, and then type a filename, an elegant implementation of speech allows the user to simply say "Save As document1" and it is done. While NaturallySpeaking gives us this power for many commands, others are too application-dependent. Even something as simple as scrolling can be implemented in many ways and it is impossible to design a system to speech enable them all. Mouse emulation gives us the ability to interact with these in a hands-free manner, but it may prove too inelegant for many users. We have shown that for some applications, like Adobe Acrobat Reader, given the development time and effort, elegant higher-level macros can be easily implemented. But others, like the IETM, have to rely on simple mouse emulation alone (although with text fill-in capability). With current operating systems and standards the way they are, these will remain application-specific.

Finally, initial testing of Dragon NaturallySpeaking shows it to be suitable for our requirements and will be able to perform satisfactorily as both a command/control recognizer and a continuous speech system for inclusion in the concept demonstration. However, as part of a Phase II effort, it may be advantageous to look into developing our own speech engines using a Hidden Markov Model development system such as Entropic Inc.'s HTK or the ISIP's ASR which is a public domain system. While costlier because of increased development time, developing our own engines would allow us to build an extremely robust, speaker-independent, small vocabulary, command/control system that has been directly trained using data files recorded by the throat microphone under the correct noise conditions. This would dramatically improve recognition, but would also limit the applicability to scenarios similar to those used during training.

In summary of our Phase I Objectives and the results of achieving them, we have demonstrated the basic feasibility of the VHIC concept--to provide a hands-free wearable computer control system for maintenance applications. The Phase I project was specifically structured to be narrow in scope, but of sufficient depth to set the foundation and provide a roadmap for a full-scope development and validation effort in Phase II. To set the stage for this Phase II effort, we next describe our specific definition of the Phase II problem which is based upon what we accomplished and learned in Phase I.

## 9.0 PHASE II RECOMMENDATIONS

Our goal for Phase II is to further develop the VHIC system into a robust, user-friendly, hands-free wearable computer control. To achieve this, system testing, redesign, and re-testing all done in close relation with the warfighter and end-user will be the most important aspect of the Phase II effort. Toward that end, we will develop and test all elements of the VHIC system in concert with a three-phase testing approach, accompanied by a prototype implementation activity:

- Laboratory Phase--Collect empirical data on individual task parts--compare and contrast elements of the system.
- Synthetic Phase--Collect empirical data on the interaction of a few elements in a controlled environment.
- Field Phase--Supplement information found in the synthetic environment by providing real-world feedback, test a synthesized system in the working environment, collect objective data, and subjective feedback.
- Implementation Activity--Produce a Phase II prototype system (this activity will parallel the testing phases) to ensure incorporation of the test findings in the evolving prototype system.

The information collection and testing (studies) in the three test phases will be the crux of our development effort, because they will provide the platform on which we will continually refine and update the system being developed in the implementation activity; and in turn, continually test the feasibility and usability of all aspects of our design. The implementation phase will ultimately finalize the system. Only through continual in-house and field testing, with experts and users alike, can we expect to specifically addresses the critical problem issues and maintain user acceptance. Our previous working relationships with the personnel and maintainers at the 178$^{th}$ Fighter Group OANG and the 445$^{th}$ C-141 Air Force Reserve Unit at WPAFB should allow us to obtain honest and candid feedback at every stage of our design process.

Cost savings associated with this approach are substantial. Savings will be realized in both testing and system design. For testing, inexpensive mechanisms can often be used in synthetic task environments. The systematic approach to critical element identification eliminates unnecessary mechanisms. Finally, the building-block nature of this approach eliminates extraneous, expensive field-testing by limiting field tests to critical issues identified in laboratory and synthetic testing. System design solutions, resulting from this approach, are also less costly because they pinpoint real system problems as proven in the laboratory or synthetic tests or as identified in the field evaluation, all while involving the end-users themselves and thus eliminating costly redesigns that can occur when a system is developed to a mature end before being introduced.

Our Phase II efforts will also be directed towards packaging and integrating the system into a compact, lightweight, wearable computer and HMD. The ability to package it into this type of display will go a long way towards user acceptance of the maintenance wearable concept. To that end, many refinements and design enhancements will be performed to take the successful Phase I concept and develop a viable product for both military and commercial applications in Phase II. These include:

- Easy installation and set-up. The software will be enhanced to allow for easy installation across platforms and operating systems.

- Packaging. The VHIC system itself consists only of the required software, the throat microphone, and the inertial head-tracker. The wearable computer platform it would run on could be one of many types. Therefore, we must design the input devices to be used with any potential wearable computer. So that as wearable designs mature and HMD profiles get smaller, the system would still be viable. This means designing mounts, cabling, and connections that can be easily installed, mounted, adjusted, and modified to fit any wearable and any HMD. This must be done in such a way as to minimize weight and obstruction to allow maintainers to use the devices in tight work areas while remaining comfortable and functional.

- Further noise dampening of throat microphone. Of primary concern for achieving the mission objective is the ability to use the device in dynamically-noisy environments such as a flight line. Further development of the throat microphone mounting and software filtering would extend the operating range of the device and reduce errors. Thus increasing the usability and acceptance of the device and further decreasing downtime by allowing operation under adverse conditions instead of having to wait for more ideal conditions.

- User customizable features. Another development for Phase II will be the ability for the user to *easily* change command vocabularies, preferred watchword, etc., without having to physically edit the vocabulary and VHIC command files. A user-friendly Windows interface will be developed to allow for customizable feature changes. This will increase user-acceptance by allowing for personal preferences, regional differences in terminology, etc.

- Development of macro-enhanced IETMs. Of further concern for Phase II is the ability to interact with IETMs that are designed by third parties (e.g., Lockheed Martin) and do not use standard Windows API commands. As part of our Phase II efforts, we will interact with these developers to establish a dialog for control interaction.

Based on the approach discussion above, we can now summarize the Phase II problem concisely--it consists of the following aspects:

- Conduct a three-phase testing effort, including laboratory, synthetic, and field phases, to evolve and refine VHIC with user participation. The test methodology must address the four primary components of a system--Software, Hardware, Environment, and Liveware (the human user).

- Produce a VHIC prototype system in an accompanying, implementation activity. Topics such as system hardware and software design, system usability in the actual maintenance environment, and overall user acceptance must be addressed in this development, accompanying the phased test methodology.

- In this testing and prototype implementation of VHIC, include the following features:

  - A vocabulary which:
    - supports multiple user strategies (A. combines the pointer and voice commands to mimic the traditional point-and-click "mouse" strategies, B. utilizes voice commands as a "direct manipulation" strategy, or C. combines context-specific voice commands with cursor movement.);
    - includes the names of on-screen hyperlinks;
    - whenever possible, takes advantage of direct input (access commonly-used functions with a minimum number of voice commands); and
    - accounts for occupation-specific slang (i.e., multiple words and phrases which might apply to the same item).

  - Capability to transparently replace the mouse and keyboard currently used for desktop applications.

  - Integration of the Gyropoint Mouse (as the mouse interface), and the Dragon NaturallySpeaking product.

  - Easy installation and setup.

  - Packaging for different types of wearable computer platforms.

  - Effective noise-dampening of throat microphones.

  - Macro-enhanced IETMs.

As a result of these studies, design improvements, developments, and packaging efforts, we will produce a fully-tested, integrated, demonstrable prototype system to form a sound foundation for Phase III or other developments, and to support our commercialization effort.

# 10.0 PHASE II TECHNICAL OBJECTIVES

## 10.1 Objective 1--Refine and Document VHIC Program Requirements and Build the Prototype

Under this objective, we will extend the requirements that were established for the Phase I Program and further focus our efforts on the functional and operational elements of our Phase II approach. This includes developing the VHIC prototype. Here, we will work closely with our AFRL sponsors and other Air Force organizations to formalize the complete requirements for VHIC. This objective will seek to answer several key questions including:

- What are the specific and detailed maintenance procedures that VHIC must support?
- What type of electronic T.O.s must VHIC operate?
- What are the vocabulary requirements for voice enabling a usable system?
- What is the most cost-effective configuration of the system?

This objective addresses all the problem aspects described in Section 2.3.2, from the definition and development perspective.

## 10.2 Objective 2--Demonstrate Final VHIC Configuration

Under this objective, we will culminate our phased prototype development and testing under Objective 1 by demonstrating the effectiveness of the VHIC system to meet the Air Force aircraft maintainers requirements. Throughout the effort, we will have SMEs use the system to accomplish actual/simulated aircraft maintenance and solicit feedback. The testing technique we will use is termed the ***Lab-Synthetic-Field (LSF)*** method. The main goal of the LSF process is to examine the system elements categorized by the ***SHEL*** model and identify any potential influences. Any or all of these elements can affect aircraft maintenance; and, this is even more likely as the elements interact.

We will continue this SME participation throughout our three-phase development approach. The first phase involves laboratory testing to collect empirical data on specific components of the task. For example, laboratory testing would be conducted on new hardware and software configurations. A laboratory test may compare two types of control input devices to see which is most compatible for interacting with maintenance technical data. The second phase of testing, the synthetic test, then adds specific variables associated with the end user environment. For example, control input devices could be specifically tested using certain characteristics of flight line maintenance such as limited reach and access. However, the synthetic test is still conducted in a controlled setting. The third and final phase involves field-testing. The purpose of a field test, as defined by this approach, is to supplement information found in the synthetic environment by providing real-world feedback about the task. In the field test, the system is tested in its working environment. While collecting objective data is possible in a field test, it is often very difficult to control extraneous factors in the actual environment. Therefore, the field test is used primarily to collect subjective feedback on the system. The final demonstration will provide explicit verification of VHIC's capabilities resulting from the phased development and testing described above. This objective addresses all the problem aspects described in Section 2.0, from the demonstration perspective.

### 10.3 Objective 3--Commercialize the VHIC System

Under this objective, we will seek to commercialize our tools and technology as set forth in our Commercialization Strategy. From previous experience, we realize that this is a significant challenge that cannot merely be paid lip service during the early stages of the Phase II Program if we are going to expect reasonable commercialization success. Hence, we have established this technical objective at the same level as the others. With this objective, we seek to answer the following questions:

- Who are the potential users of this technology?
- What features and additional capability (beyond the SBIR requirements) are necessary to improve the marketability of the system design and how much additional funding is necessary to complete the "commercial" version of the product?
- What type of support/development infrastructure is necessary to support the commercialization?

(REMAINDER OF PAGE INTENTIONALLY LEFT BLANK.)

Here, we will formalize and focus our efforts to commercialize the VHIC tools and technology. Figure 10.3-1 illustrates the Phase II schedule.

| TASK NAME | | 2001 | | | | | | 2002 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | DEC | JAN | MAR | MAY | JUL | SEP | NOV | JAN | MAR | MAY | JUL | SEP | NOV | DEC |
| **Task 1 Phase 1 System Requirements** | | ■ | ■ | | | | | | | | | | | |
| Preliminary System Specification | | | | | | | | | | | | | | |
| **Kickoff Meeting** | | ▲ | | | | | | | | | | | | |
| **Task 2 Produce Prototype** | | | ■ | ■ | | | | | | | | | | |
| Hardware Components | | | | | | | | | | | | | | |
| Software Components | | | | | | | | | | | | | | |
| Integration | | | | | | | | | | | | | | |
| **Task 3 Field-testing and Demonstration** | | | | ■ | ■ | ■ | | | | | | | | |
| **Task 4 Requirements Analysis** | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | | |
| Re-assessment | | | | | | | | | | | | | | |
| Re-design (as required) | | | | | | | | | | | | | | |
| **Task 5 Phase II Deliverables** | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| Commercialized Product Specification | | | | | | | | | | | | | ▲ | |
| Final Report | | | | | | | | | | | | | | ▲ |
| VHIC Prototype | | | | | | | | | | | | | | ▲ |

**Figure 10.3-1. VHIC Phase I Schedule**

# APPENDIX A

# TASK DATA SUMMARY

# APPENDIX B

# COMMANDS

# APPENDIX C

# CURRENT WEARABLE COMPUTER SPECIFICATIONS

# APPENDIX D

# LITERATURE REVIEW

# PDF Task Data Summary Used in the Experiment of Section 4.0

(Commands in ***Bold italic*** have been implemented in the VHIC concept demo: see Appendix B for a full list of VHIC commands)

## Button Activation Requiring Cursor Usage

| LEFT MOUSE CLICK | |
|---|---|
| **50** | ***"Select"*** |
| **31** | ***"Enter"*** |
| **24** | ***"Click"*** |
| **24** | ***"Open"*** |
| 13 | "Select pointer" |
| **4** | ***"OK"*** |

## Zooming

| GENERIC ZOOM--NEEDS CURSOR | | FIT FULL PAGE--NO CURSOR | |
|---|---|---|---|
| **7** | ***"Zoom in"*** | **7** | ***"View, Fit Visible"*** |
| **7** | ***"Magnify"*** | 5 | "Fit Page" |
| 7 | "Maximize" | 3 | "Fit full page" |
| **3** | ***"Zoom out"*** | 3 | "Fit to window" |
| 3 | "Enlarge" (pointing to it) | 3 | "Make current page fit inside window" |
| 3 | "Expand" (cursor over button) | 3 | "Make page fit window" |
| 2 | "Increase" | **2** | ***"View, Fit Page"*** |
| 2 | "Minimize" | 2 | "Show entire page" |
| 1 | "Zoom" (cursor over button) | 1 | "Enlarge page to fill entire screen" |
| | | 1 | "Fit height" |
| | | 1 | "Make fit full page" |
| | | 1 | "Show page to fill full screen" |
| **FIT PAGE WIDTH--NO CURSOR** | | **ZOOM TO SPECIFIC %--NEEDS CURSOR** | |
| 6 | "Enlarge page to width of screen" | ***10*** | ***"Zoom to XX%"*** |
| 3 | "Expand page to width of screen" | 4 | "Decrease size to XX%" |
| 3 | "Fit width to page" | 4 | "Increase size to 100%" |
| 3 | "Increase page size to fill window" | 3 | "Make page 100%" |
| 2 | "Enlarge page to fill screen" | 3 | "Select XX%" (cursor on zoom tool) |
| 2 | "Make page fill entire screen" | 1 | "Bring entire page to 75%" |
| **2** | ***"View, Fit Width"*** | 1 | "enlarge to 125%" |
| 1 | "Fit width" | 1 | "Enlarge page by 5%" |
| 1 | "Make visible width" | 1 | "Magnify table by 50%" |
| | | 1 | "View 100%" |
| | | 1 | "Reduce to 50% of current size" |
| | | 1 | "Decrease width by 50%" |

## Moving/Scrolling

| FOLLOWING PAGE--NO CURSOR | | PREVIOUS PAGE--NO CURSOR | |
|---|---|---|---|
| **32** | ***"Next page"*** | **4** | ***"Previous page"*** |
| 1 | "Move to bottom of page" | 4 | "Back" (back a page) |
| 1 | "Next" | 3 | "Go back" (a page) |
| 1 | "Next screen" | | |
| 1 | "Page 2" | | |
| 1 | "Page forward" | | |
| 1 | "Scroll a page" | | |
| 1 | "Show next page" | | |
| 1 | "Turn the page" | | |
| 1 | "Top of page" | | |

| SCROLL UP/DOWN--NO CURSOR | | SCROLL LEFT/RIGHT--NO CURSOR | |
|---|---|---|---|
| **32** | ***"Scroll down"*** (one line at a time) | **3** | ***"Scroll left"*** |
| ***13*** | ***"Page Down"*** | **3** | ***"Scroll right"*** |
| 12 | "Arrow down" | | |
| **9** | ***"Scroll up"*** | | |
| 6 | "Continue scrolling down—stop" | | |
| 5 | "Scroll to bottom of page" | | |
| 4 | "Scroll to WORD/PHRASE" | | |
| ***2*** | ***"Page up"*** | | |
| 1 | "Select arrow down" | | |
| 1 | "Move down" | | |

## SEARCH/FIND item

| WORD SEARCH | |
|---|---|
| 9 | "Go to WORD/PHRASE" |
| 9 | "Show WORD/PHRASE" |
| **5** | ***"Find WORD/PHRASE"*** |
| 4 | "Display WORD/PHRASE" |

## Highlighting

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 8 | "Highlight WORD/PHRASE" | 1 | "Highlight - Stop highlight" |
| | | 1 | "Highlight start - Highlight end" |
| | | 2 | "Highlight - End highlight" |
| | | 1 | "Highlight line" (pointing to line) |
| | | 2 | "Highlight paragraph" (pointing to ¶) |
| | | 6 | "Highlight paragraph 1-6" |
| | | 2 | "Start highlight - End highlight" |

## Link Activation

| VERBAL ONLY | |
|---|---|
| 26 | "Open LINK" |
| 19 | "Select LINK" |
| 15 | "Display LINK" |
| 9 | "Bring up LINK" |
| 8 | "Go to LINK" |
| 6 | "Show LINK" |
| 5 | "LINK" |
| 5 | "Bring up WORD/PHRASE" |

## Screen Configuration

### Navigation Pane (NP)

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 3 | "Bring up NP" | *3* | *"Select"* |
| 1 | "Display NP" | *2* | *"Click"* |
| 1 | "Open NP" | *1* | *"Open"* |
| 1 | "NP" | | |
| 1 | "Go to NP" | | |
| 1 | "View NP" | | |

### Expand Outline in NP

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 9 | "Open 'name here'" | *6* | *"Click"* |
| 3 | "Select 'name here'" | *5* | *"Select"* |
| 2 | "Display 'name here'" | 3 | "Expand" |
| 1 | "name here" | *2* | *"Enter"* |
| | | *2* | *"Open"* |

## Close Program

| VERBAL ONLY | | REQUIRES CURSOR USAGE ON BUTTON/LINK | |
|---|---|---|---|
| 3 | "Close Program" | *3* | *"Close"* |
| 2 | "Exit" | *2* | *"Select"* |
| | | *2* | *"Enter"* |
| | | *1* | *"Click"* |

# IETM Task Data Summary

## Close Window

| VERBAL ONLY | REQUIRES CURSOR USAGE ON BUTTON/LINK |
|---|---|
| 38 "Close"<br>3 "Remove Figure"<br>3 "Exit"<br>1 "Figure 1 off" | ***5 "Click" (Close box)***<br>***3 "Select" (Close box)***<br>***3 "Enter" (Close box)***<br>***1 "Double Click" (Close box)*** |
| **REQUIRES CURSOR USAGE ON PULL-DOWN MENU** | |
| ***"Select" (File Menu) → "Select" (Close)*** | |

## Close Program

| VERBAL ONLY | REQUIRES CURSOR USAGE ON BUTTON/LINK |
|---|---|
| 6 "Exit"<br>2 "Select exit"<br>1 "Press exit"<br>1 "Exit all" | ***1 "Select" (button)***<br>***1 "Open" (button)***<br>***1 "Enter" (button)***<br>1 "Click on that" (button) |

## Zooming

| VERBAL ONLY | REQUIRES CURSOR USAGE OVER AREA OF INTEREST |
|---|---|
| 3 "Maximize"<br>1 "Enlarge right side of figure"<br>1 "Enlarge figure to size of screen"<br>1 "Full screen with this figure (NA)"<br>1 "Select maximize"<br>1 "Maximize window" | 5 "Zoom in"<br>4 "Enlarge Figure"<br>2 "Magnify figure"<br>2 "Expand Figure"<br>***2 "Click"*** (Center of Figure)<br>1 "Zoom figure 4"<br>1 "Magnify"<br>1 "Larger"<br>1 "Enlarge at cursor" (On first item)<br>1 "Click on that" (Maximize button)<br>1 "Zoom on that" |

## SEARCH/FIND *item*

| VERBAL ONLY | | REQUIRES CURSOR USAGE |
|---|---|---|
| 7 | "Bring up *item*" | <Not applicable> |
| 4 | "Go to *item*" | |
| 3 | "Show *item*" | |
| 2 | "Search *item*" | |
| 1 | "Find *item*" | |
| 1 | "Search" | |
| 1 | "Open *item*" | |
| 1 | "Display *item*" | |
| 1 | "Take me to *item*" | |

## Moving/Scrolling

### Bottom of the page

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 6 | "Scroll to bottom of page" | 1 | "Click hold to bottom" (Down arrow) |
| 3 | "Scroll to end" | 1 | "Select-hold…release" (Down arrow) |
| 2 | "End page" | ***1*** | ***"Enter" (Down arrow)*** |
| 2 | "Page down" (repeatedly) | 1 | "Enter…stop" (down arrow) |
| 2 | "Last page" | | |
| 2 | "Bottom of page" | | |
| 1 | "Go to bottom of page" | | |
| 1 | "Arrow down to bottom of page" | | |
| 1 | "Arrow to bottom of page" | | |
| 1 | "Go to end" | | |

### Following Page

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 2 | "Select next" | ***1*** | ***"Select" (button)*** |
| 2 | "Next page" | ***1*** | ***"Open" (button)*** |
| 1 | "Show next page" | ***1*** | ***"Enter" (button)*** |
| 1 | "Next" | 1 | "Click on that" (button) |
| 1 | "Go to next page" | ***1*** | ***"Click" (button)*** |
| 1 | "Go to next" | | |
| 1 | "Click on next" | | |
| 1 | "Next screen" | | |

## Scroll up/down

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 7 | "Scroll down" (goes down one line) | ***1*** | ***"Select" (button)*** |
| 3 | "Scroll down… stop" (continuous) | 1 | "Single Click" (Scroll Bar), |
| 3 | "Page Down" | | |
| 2 | "Arrow down" (goes down one line) | | |
| 2 | "Continue to Scroll Down" → "Stop" | | |
| 1 | "Arrow down/stop" (continuous) | | |
| 1 | "Hit arrow down three times" | | |
| 1 | "Screen at a time" | | |
| 1 | "Page at a time" | | |

## Highlighting

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 17 | "Highlight *word or phrase*" | 7 | "Highlight line" |
| 4 | "Select *word or phrase*" | 5 | "Begin highlight…End highlight" |
| | | 2 | "Highlight startpoint….endpoint" |
| | | 2 | "Highlight from here…to here" |
| | | 1 | "Highlight…..end highlight" |
| | | 1 | "Select….complete" |
| | | 1 | "Click and highlight to end of line" |
| | | 1 | "Highlight…enter" |

## Button Activation

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 42 | "<*button name*>" | ***21*** | ***"Click"*** |
| 22 | "Select <*button name*>" | ***18*** | ***"Select"*** |
| 12 | "Bring up <*button name*>" | ***18*** | ***"Open"*** |
| 12 | "Go to <*button name*>" | ***17*** | ***"Enter"*** |
| 12 | "Show <*button name*>" | ***1*** | ***"Double click"*** |
| 4 | "Press <*button name*>" | | |
| 1 | "Check <*button name*>" | | |
| 1 | "Open <*button name*>" | | |

## Link Activation

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 32 | "Open *link*" | ***25*** | ***"Select"*** |
| 16 | "Bring up *link*" | ***25*** | ***"Enter"*** |
| 14 | "Go to *link*" | ***17*** | ***"Click"*** |
| 14 | "Select *link*" | ***5*** | ***"Open"*** |
| 10 | "*link*" | ***4*** | ***"Double click"*** |
| 10 | "Show *link*" | 3 | "Show" |
| 2 | "Display *link*" | 1 | "Bring up" |

## Return to Main Menu

| VERBAL ONLY | | REQUIRES CURSOR USAGE | |
|---|---|---|---|
| 15 | "Main menu" | ***11*** | ***"Select"*** |
| 8 | "Back to main menu" | ***9*** | ***"Enter"*** |
| 7 | "Go to main menu" | ***8*** | ***"Open"*** |
| 4 | "Select main" | 6 | "Click on that" |
| 4 | "Go back to main menu" | ***5*** | ***"Click"*** |
| 4 | "Bring up main menu" | 1 | "Click button, select." |
| 4 | "Show main menu" | 1 | "Single Click" |
| 3 | "Go back" | | |
| 1 | "Return to main menu" | | |
| 1 | "Back" | | |
| 1 | "Select main menu" | | |

## Task specific

| INITIATING HELP | | BOOKMARK | |
|---|---|---|---|
| "Bring up instructions on how to create bookmarks" | | 2 | "Bookmark *word or phrase*" |
| "How do I set up a bookmark?" (NA) | | 2 | "Create Bookmark *word or phrase*" |
| "Go to instructions for annotations" | | 1 | "Insert Bookmark at Figure *X*" |
| "How to create annotation" | | | |
| **ANNOTATION** | | | |
| 4 | "Create annotation *word or phrase*" | | |
| 2 | "Annotate *word or phrase*" | | |

# VHIC COMMANDS

(Available in *Limited Command Mode*, *Command Mode* and *Dictation Mode*)

***Complete List of Available Commands:***

## Selection Commands

| SPEECH | RESULT |
|---|---|
| "Double click" | Double clicks the left mouse button at the current cursor position. |
| "Left click"<br>"Click" | Single clicks the left mouse button at the current cursor position. This command can be said while the microphone is asleep or while VHIC is in watchword mode. |
| "Drag"<br>"Left button down" | Presses and holds the left mouse button at the current cursor position. |
| "Drop"<br>"Left button up" | Releases the left mouse button. |
| "Right click" | Single clicks the right mouse button at the current cursor position. |
| "Middle click" | Single clicks the middle mouse button at the current cursor position. |

## Pointing Commands

| SPEECH | RESULTS |
|---|---|
| "Move mouse slower" | Decreases cursor speed relative to the amount of movement by the mouse or head-tracker (not available using Win95). |
| "Move mouse faster" | Increases cursor speed relative to the amount of movement by the mouse or head-tracker (not available using Win95). |
| "Reset mouse speed" | Resets cursor speed to its default value (not available using Win95). |
| "Slowest mouse speed" | Decreases cursor speed to a minimum (not available using Win95). |
| "Mouse left" | Starts the cursor moving left. |
| "Mouse right" | Starts the cursor moving right. |
| "Mouse up" | Starts the cursor moving up. |
| "Mouse down" | Starts the cursor moving down. |
| "Faster" | Moves the cursor faster. |
| "Slower" | Moves the cursor slower. |
| "Stop mouse" | Stops the cursor movement. |
| "Center mouse" | Moves the cursor to the middle of the screen. |
| "Go left" | Moves the cursor left a small distance. |
| "Go right" | Moves the cursor right a small distance. |
| "Go up" | Moves the cursor up a small distance. |
| "Go down" | Moves the cursor down a small distance. |

## Scrolling Commands

| SPEECH | RESULT |
|---|---|
| "Scroll line up" | Scrolls up a line. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll line down" | Scrolls down a line. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll page up" | Scrolls up a page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll page down" | Scrolls down a page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll left" | Scrolls left by one unit. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll right" | Scrolls right by one unit. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll page right" | Scrolls right one page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Scroll page left" | Scrolls left one page. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling left" | Starts automatic scrolling to the left. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling right" | Starts automatic scrolling to the right. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling up" | Starts automatic scrolling up. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Start scrolling down" | Starts automatic scrolling down. The cursor must be placed within the window to scroll and that window must have a scroll bar. Not all windows with scroll bars can be scrolled using speech. |
| "Stop scrolling" | Stops automatic scrolling. |
| "Speed up" | Makes automatic scrolling scroll faster. |
| "Slow down" | Makes automatic scrolling scroll slower. |

## Zooming Commands

| SPEECH | RESULT |
|---|---|
| "Zoom in"<br>"Half scale" | Zooms in at the current cursor position. This command is only valid if Acrobat Reader is open and has focus. |
| "Zoom out"<br>"Double scale" | Zooms out from the current cursor position. This command is only valid if Acrobat Reader is open and has focus. |

## Paging Commands (*Acrobat Reader only*)

| SPEECH | RESULT |
|---|---|
| "Next page" | Displays the next page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "Previous page" | Displays the previous page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "First page" | Displays the first page of a document. This command is only valid if Acrobat Reader is open and has focus. |
| "Last page" | Displays the last page of a document. This command is only valid if Acrobat Reader is open and has focus. |

## Search Commands (*Acrobat Reader only*)

| SPEECH | RESULT |
|---|---|
| "Find"<br>"Find word" | Opens the find dialog box. This command is only valid if Acrobat Reader is open and has focus. |
| "Find again"<br>"Find next" | Resumes searching for more occurrences of the phrase from the previous search. This command is only valid if Acrobat Reader is open and has focus. |
| "Go"<br>"Start search" | Begins searching for the words entered into the find dialog box. This command is only valid if the Acrobat Reader find dialog box is open and has focus. |
| "Match whole word only" | Toggles the "Match Whole Word Only" check box of the find dialog box. This command is only valid if the Acrobat Reader find dialog box is open and has focus. |
| "Match case" | Toggles the "Match Case" check box of the find dialog box. This command is only valid if the Acrobat Reader find dialog box is open and has focus. |
| "Find backwards" | Toggles the "Find Backwards" check box of the find dialog box. This command is only valid if the Acrobat Reader find dialog box is open and has focus. |
| "Find what" | Gives keyboard focus to the "Find What" field of the find dialog box. This command is only valid if the Acrobat Reader find dialog box is open and has focus. |

## Mode Commands

| SPEECH | RESULT |
|---|---|
| "Watchword mode" | Begins watchword mode. The phrase "computer" or "wake up" must precede any speech command. Exception: "Left click" is a valid command during watchword mode. |
| "Quit watchword mode" | Ends watchword mode. |
| "Dictation mode" | Turns on global dictation. When global dictation is on, any Windows control that accepts text can be dictated into using speech. All commands are still recognized as well but with limited accuracy. |
| "Command mode" | Returns to command mode and turns global dictation off. In command mode only built in NaturallySpeaking commands and VHIC commands are recognized. Other applications that have been specifically programmed to be dictated into by Naturally Speaking will still accept dictation. |
| "Limited command mode" | In limited command mode only VHIC commands are recognized (Global dictation is off and built-in NaturallySpeaking commands are off). Limited command mode will work exactly as Command Mode does if another application is using Naturally Speaking and has activated the built-in commands (e.g., if Microsoft word is open and has Natural Word enabled). |

## Note Commands

| SPEECH | RESULT |
|---|---|
| "Take note" | Opens a note box that can be dictated into. |
| "Save note" | Saves the contents of the note box to a file and closes the note box. The file is named after the current date (ex. 091500.txt for notes saved on Sept. 15, 2000) in the directory \VHIC\Notes. |
| "Quit note" | Closes the note box without saving the note. Any text written to the note box is lost. |

## Other Commands

| SPEECH | RESULT |
|---|---|
| "Playback" | Plays back the last utterance that was recorded, skipping any utterance that was recognized as the word "Playback". |

## Part Order Form Commands

| SPEECH | RESULT |
|---|---|
| "Order part" | Starts the application OrderPart.exe bringing up the Part Order Form window. |
| "Part number <Part_Number>"<br><br>Example:<br>"Part number six one three dash two seven."<br><br>Form:<br><Part_Number> = <Digit> <Digit> <Digit> "dash" <Digit> <Digit><br><br><Digit> is one of the following: "zero", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine".<br><br>**Note**: Valid part numbers that have corresponding descriptions are 613-27, 111-11, and 123-45. | Fills in the Part Number field with the specified <Part_Number>. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Quantity <Quantity>"<br><br>Example:<br>"Quantity four" or "Quantity one three".<br><br>Form:<br><Quantity> = <Digit> or <Digit><Digit><br><br><Digit> is one of the following: "zero", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine". | Fills in the Quantity field with the specified <Quantity>. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |

| SPEECH | RESULT |
|---|---|
| "Requestor I D <Requestor>"<br>"Requestor <Requestor>"<br><br>Example:<br>"Requestor I D Grigsby"<br><br><Requestor> must be one of the following "Grigsby", "LaDue", "Valiton", Smith". | Fills in the Requestor ID field with the specified <Requestor>. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Priority <Priority_Number>"<br><br>Example:<br>"Priority one" or Priority<br><br><Priority_Number> is one of the following: "one", "two", or "three". It can be preceded by "oh" or "zero" | Checks the radio button specified by <Priority_Number>. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Send confirmation" | Toggles the Send Confirmation check box. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Print order" | Toggles the Print Order check box. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Accept" | Clicks the Accept button. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |
| "Cancel" | Clicks the Cancel button. This command is only valid if the Part Order Form window of the OrderPart.exe application is open and has focus. |

## Relevant *NaturallySpeaking* Commands

(See Appendix A of the *Dragon NaturallySpeaking User's Guide* for a complete list)
(Available in *Command Mode* and *Dictation Mode*; not available in *Limited Command Mode*)

### Selection Commands

| SPEECH | RESULT |
|---|---|
| "Click <button or menu name>" | Activates any button or menu item in the active window. Examples: "Click OK," "Click Cancel," "Click File," etc. |
| "Press <key name>" | Duplicates activation of a keyboard keypress. Examples: "Press Tab," "Press h," "Press Shift F7," etc. |
| "Cancel" | Closes a menu or undoes a previous command |

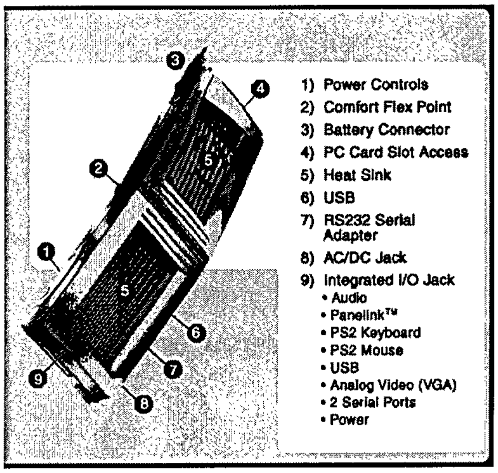
### Editing Commands

| SPEECH | RESULT |
|---|---|
| "Scratch that" | Erases the last utterance in a text (dictation) box |
| "Backspace"<br>"Backspace <#>" | Deletes the last character in a text box or the last # of characters. Example: "Backspace 5" deletes the last 5 characters. |

# CURRENT WEARABLE COMPUTER SPECIFICATIONS

## ViA II PC

ViA Incorporated, Burnsville, MN
(*www.flexipc.com*)




| | |
|---|---|
| **Processor**: | Cyrix Media Gxi @ 166 MHz |
| **Chipset**: | Cyrix CX5520 |
| **Memory**: | 64 MBytes EDO RAM |
| **Hard Disk**: | IBM 3GN TravelStar, 3.2 GBytes, 2.5" ultra-low profile HDD |
| **PC Card**: | Two type II cards or one type III card, Cardbus support |
| **USB**: | USB 1.0 connector |
| **Serial**: | RS-232 mini connector |
| **Display**: | various |
| **Cost**: | N/A |

## Xybernaut Mobile Assistant IV

Xybernaut Corp., Fairfax, VA
(*www.xybernaut.com*)

| | |
|---|---|
| **Processor**: | Pentium MMX @ 233 MHz |
| **Chipset**: | Intel |
| **Memory**: | 160 MBytes RAM |
| **Hard Disk**: | 4.3 GBytes HDD (up to 8 GBytes avail.) |
| **PC Card**: | Two type II cards or one type III card, Cardbus support |
| **USB**: | USB 1.0 connector |
| **Serial**: | RS-232 mini connector |
| **Display**: | Xyberview HMD (640x480 color) |
| **Cost**: | $6995 (as listed above) |

## LITERATURE REVIEW

Acero, A. and Stern, R.M. (1990). Robust speech recognition by normalization of the acoustic space. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing* (Albuquerque, NM), 849-852.

Adams, K.D. and Bentz, B.D. (1995). Headpointing: the latest and greatest. *Proceedings of the RESNA 18th Annual Conference* (Vancouver, Canada), 446-448.

Amon, C.H., Finger, S., Gursoz, L., Prinz, F., Siewiorek, D.P., Smailagic, A., Stivoric, J., and Weiss, L. (1994). Reflections on a Concurrent Design Methodology: A Case Study in Wearable Computer Design. *Computer Aided Design, Special Issue on Concurrent Design.*

Amon, C.H., Nigen, J.S., Siewiorek, D.P., Smailagic, A., and Stivoric J. (1994). Concurrent Design and Analysis of the Navigator Wearable Computer System: The Thermal Perspective. *IEEE Trans. on Components, Hybrids, and Manufacturing Technology 17*, No.6.

Anderson, T.R. (1998). The technology of speech-based control. In *Alternative Control Technologies: Human Factors Issues.* RTO Lecture Series 215 (RTO-EN-3 AC/323(HFM)TP/1).

Andres, R.O. and Hartung, K.J. (1989). Predictions of head movement time using Fitts' law. *Human Factors 31*, 703-713.

Angelo, J., Deterding, C., and Weisman, J. (1991). Comparing three head-pointing systems using a single subject design. *Assistive Technology 3*, 43-49.

Barry, T.P., Liggett, K.K., and Williamson, D.T. (1999). Human-electronic crew communication: Applications for speech recognition in the cockpit. In J. Reising, R. Taylor & R. Onken (Eds.), *The Human-Electronic Crew: The Right Stuff?* Wright-Patterson AFB OH: Air Force Research Laboratory, Human Effectiveness Directorate, Crew System Interface Division (AFRL-HE-WP-TR-1999-0235), 123-129.

Bass, L., Kasabach, C., Martin, R., Siewiorek, D., Smailagic, A., and Stivoric, J. (1997). The design of a wearable computer. In *Companion Proceedings of the CHI '97 Conference on Human Factors in Computing Systems*, New York: Association for Computing Machinery, pp. 139-146.

- explains the design process used for the dial design of the CMU VuMan3 system. The authors stress the need for an integrated concept that permeates the project as a key to success. However, it should be noted that the software is system and application dependent and therefore limits the use of the controller and interface.

Baudin, S.E. (1996). Telemaintenance applications for the wearable PC. *Proceedings of 15th AIAA/IEEE Digital Avionics Systems Conference*, 407-413.

Beringer, D.B. and Scott, J. (1985). The long-range light pen as a head-based user-computer interface: Head-mounted "sights" versus head positioning for computer access by the disabled. *Proceedings of the Human Factors Society 29th Annual Meeting*, 114-118.

Borah, J. (1989). *Helmet-Mounted Eye Tracking for Panoramic Display Systems – Volume I: Review of Current Eye Movement Measurement Technology* (AAMRL-TR-89-019) (NTIS: AD-A273 101/6/XAB).

Borah, J. (1995). *Investigation of Eye and Head Controlled Cursor Positioning Techniques.* AL Tech. Report #AL/CF-SR-1995-0018

- Head control of cursor position is viable over a wide range target sizes and motion distances. While not as fast as mouse control, it is comparable to pure eye control (for larger targets) and superior to a eye/head hybrid over most of the ID range.

Borah, J. (1998). Technology and application of head based control. In *Alternative Control Technologies: Human Factors Issues.* RTO Lecture Series 215 (RTO-EN-3 AC/323(HFM)TP/1).

Brady, M., Doumas, T., and Vanderheiden, G. (1984). Two headpointing input techniques – A comparative study. In *Proceedings of the 2nd International Conference on Rehabilitation Engineering*, Ottawa, 48-49.

Bruegge, B. and Bennington, B. (1996). Application of wireless research to real industrial problems: Applications of mobile computing and communication. *IEEE Personal Communications 3*, 64-71.

Calhoun, G.L. and McMillan, G.R. (1998). Hands-free input devices for wearable computers. *Proceedings of the Fourth Annual Symposium on Human Interaction with Complex Systems*, 118-123.

Chapman, D. and Simmons, J. (1995). A comparative evaluation of voice versus keypad input for manipulating electronic technical data for flightline maintenance technicians. *Master's Thesis*, Air Force Institute of Technology, Wright-Patterson, AFB, Ohio, 1995.

- Task should be selected based on the degree to which tech data must be used due to task complexity, and the degree to which both hands are used to complete the task. Tasks such as a weapons load task, a hydraulic pump change, a flight control troubleshooting task, or a tire change would be appropriate

Clark, M. R. (1996). Frontiers in User Interface Design: Wearable Computers. In *Proceedings of the Stereoscopic Displays and Virtual Reality Systems III*, Bellingham, WA: SPIE, 302-305.

Curry, M., Hobbs, J., and Toub, R. (1996). Will using a head mounted display affect pointing on wearable computers? Submitted for consideration in *2nd ACM International Conference on Mobile Computing and Networking* (MobiCom 96), White Plains, NY, November 11-12, 1996. (http://www.cs.uoregon.edu/research/wearables/Papers/AdvUIpaper.html)

- Compares performance for a mouse pointing task using a large screen projection display, an immersive HMD, and a see-through HMD. The paradigm was a typical Fitts' Law pointing task (click to start, move cursor inside target, click to confirm). No significant performance differences were found for the three conditions (display types). All gave similar movement times and error rates. The authors conclude "this study presents evidence that suggests that the HMD is at least as good as traditional desktop monitors for pointing tasks in both speed and accuracy." However, only mouse pointing was used, it remains to be seen how head pointing and HMDs interact.

Department of Defense (1992a). Data base. revisable: Interactive Electronic Technical Manuals, For the support of (MIL-D-87269). Washington, DC: Department of Defense. (NTIS No. PB93-145225/XAB)

Department of Defense (1992b). Manuals. Interactive Electronic Technical: General content, style, format, and user-interaction requirements (MIL-M-87268). Washington, DC: Department of Defense. (NTIS No. PB93-144616/XAB)

Department of Defense (1992c). Quality assurance program: Interactive Electronic Technical Manuals and associated technical information, Requirements for (MIL-Q-87270). Washington, DC: Department of Defense. (NTIS No. PB93-144608/XAB)

Edwards, E. (1988). Introductory overview. In E.L. Weiner & D.C. Nagel (Eds.), *Human Factors in Aviation.* San Diego, CA: Academic Press.

Evans, G. and Blenkhorn, P. (1999). A head operated joystick – Experience with use. *CSUN_99 workshop.* (http://www.dinf.org/csun_99/session0019.html).

- The authors developed an inexpensive LED-based joystick-style head tracking device for use by disabled users. They then compared this device to absolute positioning trackers (standard head trackers). The results show: (1) for many users, the joystick style of interaction is preferable to the absolute positioning style because it requires less accurate head movements which are easier for disabled users; (2) users can tolerate inaccuracies in joystick-style devices in a transparent manner, and (3) because of the reduced requirements for accuracy, joystick devices may be produced and marketed at lower cost.

Friend, J.A. and Grinstead, R.S. (1992). Comparative evaluation of a monocular head mounted display device versus a flat screen display device in presenting aircraft maintenance technical data. *Master's thesis*, AFIT/GSM/LSM/92S-8. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH (AD-A259684).

- Tasks should include activities requiring in-depth troubleshooting analysis, activities requiring fast response turnaround, activities requiring schematics, and complex testbeds.

Head-Mounted Display Survey (1998). *Real Time Graphics 7* (#2), 1-2, 8-12.

Hemphill, C.T. and Thrift, P.R. (1995). Surfing the web by voice. In *Proceedings of the ACM Multimedia '95*, San Francisco, CA. ACM, November, 1995.

*Input Devices: There Is No Holy Grail*
(http://wearables.www.media.mit.edu/projects/wearables/input-guidelines.html)

- find speech interaction for wearables is a problem "during conversation or conference or whenever privacy is required"
- don't like handwriting input (a la PDAs)
- believe that chording keyboards (Twiddler) are the answer [however, these are obviously not hands free]

Jagacinski, R.J. and Monk D.L. (1985). Fitts' law in two dimensions with hand and head movements. *Journal of Motor Behavior 17*, 77-95.

Kancler, D.E., Revels, A.R., Quill, L.L. and Nemeth, K. (1999). *SBIR Phase II: GMD Baseline Test Comparing the GMD with Traditional Displays* (Contract No. F41624-98-C-5004, Project No. 6004). Dayton, OH: University of Dayton Research Institute.

Kim, D., Richards, S. W., and Caudell, T. P. (1997). An Optical Tracker for Augmented Reality and Wearable Computers. In *Proceedings of the IEEE 1997 Virtual Reality Annual International Symposium*, Los Alamitos, CA: IEEE Computer Society Press, 146-151.

Koons, D. B., Sparrell, C. J., and Thorisson, K. R. (1993). Integrating Simultaneous Input from Speech, Gaze, and Hand Gestures. In M. T. Maybury (Ed.), *Intelligent Multimedia Interfaces*, Menlo Park, CA: AAAI Press/The MIT Press, 257-276.

Lai, J. and Vergo, J. (1997). MedSpeak: Report creation with continuous speech recognition. *Proceedings of CHI '97* (Atlanta, GA), ACM Press, 431-438.

Landay, J. A. and Kaufmann, T. R. (1993). User Interface Issues in Mobile Computing. In *Proceedings of the Fourth Workshop on Workstation Operating Systems*, Los Alamitos, CA USA: IEEE, 40-47.
(http://www.cs.cmu.edu/~landay/research/publications/MobileInterfaces.abstract)

Leyzerovich, A. (1996). Efficient data management for the prototype of the wearable F-16 maintenance system. *Master's Thesis.* Department of electronic and computer engineering, Information Networking Institute (INI), Carnegie Mellon University.

Liggett, K.K., Kustra, T.W., Reising, J.M., and Hartsock, D.C. (1997). A comparison of touch, manual, and head controllers for cursor movement. *Proceedings of the Human Factors and Ergonomics Society 41st Annual meeting*, 80-84.

- Found that a laser based head tracker was slower than touch screen or target designation control (TDC) on throttle but conclude that these times were due to problems with the equipment and not with head performance *per se*. They conclude "with further refinements, (head tracking) has the potential of becoming as viable and intuitive as touch."

Lin, M.L., Radwin, R.G., and Vanderheiden, G.C. (1992). Gain effects on performance using a head-controlled computer input device. *Ergonomics 35*, 159-175.

- The purpose of this study was to use Fitts' Law to determine the optimal gains for both mouse and head pointing on fixed monitors. The authors also investigated control-display gain interactions with other task factors including target width, movement amplitude, and direction. A discrete target acquisition task with multiple target sizes, target distances and meridians was used. Four display gains were used for each condition. "The results indicated that gain had a significant effect on movement time for both types of pointing devices and exhibited local minimums." For mouse movements, a gain between 1.0 and 2.0 resulted in the fastest movement times and smallest RMS cursor deviation. For head movements, minimums were attained at gains between 0.3 and 0.6. For the head-controlled pointer "(g)ain had the greatest effect for small target widths and long movement amplitudes…"

Masquelier, B.L. (1991). Comparative evaluation of monocular display devices relative to portable computers for display of aircraft maintenance technical data. *Master's thesis*, AFIT/GSM/LSM/91S-18. School of Systems and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH (AD-B161858).

- Mobile tasks which require the hands to be free, such as a flight line maintenance task. Appropriate tasks include weapons loading, engine bay tasks, hydraulics repair, end-of-runway tasks, and other checklist tasks.

Matias, E., MacKenzie, I. S., and Buxton, W. (1996). A wearable computer for use in microgravity space and other non-desktop environments. *Companion of the CHI '96 Conference on Human Factors in Computing Systems*, New York: ACM, 69-70.

McMillan G.R., Calhoun G.L., Masquelier B.L., Grigsby S.S., Quill L.L., Kancler D.E., Nemeth K., and Revels A.R. (1999). Comparison of hands-free versus conventional wearable computer control for maintenance applications. *Human Factors and Ergonomics Society Annual Meeting '99.*

McMillan, G.R., Eggleston, R.G., and Anderson, T.R. (1997). Nonconventional controls. In G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics* (2nd ed.). New York: John Wiley & Sons.

Miller, G. (1992). Voice recognition as an alternative computer mouse for the disabled. *Proceedings of the RESNA 15th Annual Conference* (Washington, DC), 55-57.

Mizell, D., Caudell, T., Segall, Z., Siewiorek, D., and Starner, T. (1996). *Boeing Wearable Computer Workshop Breakout Session Summary* (Online Report): Seattle, WA: Boeing. (http://www.cs.cmu.edu/~wearable/boeing/index.html)

Morton, S. and Bukhres, O. (1997). Mobile Computing in Military Ambulatory Care. In *Proceedings of the CBMS '97: the 10th IEEE Symposium on Computer-Based Medical Systems*, Los Alamitos, CA, USA: IEEE. (http://www.cs.iupui.edu/~smorton/MortonRes3.html)

Najjar, L.J., Thompson, J.C., Ockerman, J., and Treanor, C.J. (1996). Using a wearable computer for industrial data collection and performance support. Paper presented at the Workshop on Wearable Computer Systems meeting, Renton, WA. (http://mime1.gtri.gatech.edu/MiME/papers/Boeing.html).

Najjar, L.J., Thompson, J.C., and Ockerman, J.J. (1997). Using a wearable computer to improve the performance of quality assurance inspectors in a food processing plant. Paper presented at the CHI'97 Workshop on Wearable Computing, Atlanta, GA. (http://mime1.gtri.gatech.edu/MiME/papers/CHI'97_position_paper)

Najjar, L. J., Thompson, J. C., Ockerman, J., and Treanor, C. J. (1996). Using a wearable computer for industrial data collection and performance support. Paper presented at the *Workshop on Wearable Computer Systems* meeting, Renton, WA.

Novick, D.G. and House, D. (1995). Spoken language access to multimedia (SLAM): A multimodal interface to the world wide web. (http://www.cse.ogi.edu/SLAM/slam-paper.html).

Ockerman, J. J., Najjar, L. J., & Thompson, J. C. (1997). Evaluation of a wearable computer performance support system. In T. Muldner & T. C. Reeves (Eds.), *Educational Multimedia/Hypermedia and Telecommunications, 1997*. Charlottesville, VA: Association for the Advancement of Computing in Education, 788-793.

Ockerman, J.J. and Pritchett, A.R. (1998). Preliminary investigation of wearable computers for task guidance in aircraft inspection. *International Conference on Human-Computer Interaction in Aeronautics*, Montreal, Canada.

- Found that user's tend to have an over reliance on computer presented information when performing an aircraft inspection task. While the wearable system helped user's remember tasks they may have forgotten, it also caused them to rely more heavily on the system and only perform tasks that they were instructed to do in stead of using the computer for guidance and expanding on it. Control pilots allowed to do a free inspection often found problems that the computer did not instruct the user to look for. However, the control pilots also missed some problems the computer users did not. This result shows the importance of having a thorough and complete data set on the system.

Pausch, R., Shackelford, M. A., and Proffitt, D. (1993). A User Study Comparing Head-Mounted and Stationary Displays. In *Proceedings of the IEEE Symposium on Research Frontiers in VR*, Los Alamitos, CA: IEEE Computer Society Press, 41-45. (http://www.cs.cmu.edu/~stage3/publications/93/conferences/IEEEsymposiumOnResearchFrontiersInVR/handVsHeadUserStudy/paper.html)

Pentland, A.P., Petrazzuoli, M, Gerega, A., and Starner, T. (1997). The digital doctor: An experiment in wearable telemedicine. *Proceedings of the First International Symposium on Wearable Computers*, 173-174.

Radwin, R.G., Vanderheiden, G.C., and Lin, M.L. (1990). A method for evaluating head-controlled computer input devices using Fitts' Law. *Human Factors 32*, 423-438.

- Compares head vs. hand (mouse) controlled movements using a standard fixed monitor. A Fitts' Law paradigm was used without discrete input. Subjects had to hold the cursor in a starting position for 2 sec then move the cursor and keep it inside the target perimeter for 62.5 ms. Two target distances (24.4 and 110.9 mm) and 3 target diameters (2.7, 8.1, and 24.2 mm) were used along 8 meridians (0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°). Performance measures included movement time, cursor path distance, and root-mean-square cursor deviation. For normals, results include: average movement time for 10 subjects was 63% greater for head versus mouse; no significant effect of meridian was found for mouse but average movement time for head was lowest at 90° and 270° (vertical motions; flexion and extension), increased for diagonal movements, and was highest for horizontal movements. Accuracy (path distance and RMS deviations) was lowest for off-diagonal movements (0°, 90°, 180°, and 270°). [Further conclusions are reached regarding studies using movement impaired subjects]. *{NOTE: asymptotic performance for head pointing was achieved after ~15 consecutive sets of 48 trials}*

Revels, A.R., Kancler, D.E., Quill, L.L. and Nemeth, K. (1999). *SBIR Phase II: Test #2, GMD Mobility Test Using the GMD in a Synthetic Mobile Environment* (Contract No. F41624-98-C-5004, Project No. 6004). Dayton, OH: University of Dayton Research Institute.

Revels, A.R., Quill, L.L., Kancler, D.E., and Masquelier, B.L. (1998). Human interaction with wearable computer systems: A look at glasses-mounted displays. In *AeroSense '98* (Orlando, FL), SPIE, International Society of Optical Engineering.

Rodman, R.D. (1999). *Computer Speech Technology.* Boston: Artech House.

Ross, D. A. and Sanford, J. A. (1997). The Wearable Computer as a Remote Interface for People with Disabilities. In *Proceedings of the International Symposium on Wearable Computers*, Los Alamitos, CA, USA: IEEE Computer Society, 161-162.

Rudnicky, A.I. and Hauptmann, A.G. (1992). Multi-modal interaction in speech systems. In Blattner, M. and Dannenberg, R. *Multimedia interface design.* New York: ACM, 147-182.

Rudnicky, A.I., Reed, S.D., and Thayer, E.H. (1996). SpeechWear: A mobile speech system. (http://www.speech.cs.cmu.edu/air/papers/speechwear/speechwear.html)

Schaab, J.A., Radwin, R.G., Vanderheiden, G.C., and Krogh Hansen, P. (1996). A comparison of two control-display gain measures for head-controlled computer input devices. *Human Factors 38*, 390-403.

- Two gain measures were compared using a discrete target acquisition task. The gain measures were defined as: angle/angle (A/A) gain (the ratio between the angle subtended by displacement of the cursor and the corresponding angle of head rotation) and displacement/angle (D/A) gain (the ratio between the linear displacement of the cursor on the screen and the corresponding angle of head rotation). For the smallest viewing distance, A/A and D/A gain gave equivalent performance (ave movement time, RMS deviation), however, an increase in viewing distance led to a decrease in performance when using A/A gain but not for D/A gain. Optimal performance occurred with a D/A gain of 0.5 to 0.67 cm/deg.

Siegel, J. and Bauer, M. (1997). A field usability evaluation of a wearable system. *International Society for Wearable Computing Annual Meeting.*

- report of a field usability test of a maintenance wearable system. The main controller used was a one-hand dial device developed at CMU. The authors precluded speech because "The commercial products did not perform as advertised and their instability exceeded acceptable levels for use." However, they found problems with the dial controller – inadvertent activations caused by the user leaning the system on the aircraft – but the one handed control was preferred over two handed and they state "A completely hands-free system (e.g., speech) would probable provide additional advantages, but we would have to weigh the ease of use of the interface against the viability of use given the noise levels from other aircraft on the flightline." Also, it took the maintainers an average of 42 minutes to complete the two tasks using the wearable system and only 28 minutes without the system. This was due to a combination of controller problems, system interface problems, familiarity, and because the maintainers seemed to be more thorough when using the wearable because they followed the software step by step.

Siegel, J., Kraut, R.E., John, B.E., and Carley, K.M. (1995). An empirical study of collaborative wearable computer systems. CHI '95 Proceedings (http://www.acm.org/sigchi/Electronic/documnts/shortppr/js_bdy.html)

Siewiorek, D.P. and Smailagic, A. (1994). Software Development for the Navigator Wearable Computers. *Proceedings of the Portable by Design Conference*. Penton Publishing.

Siewiorek, D.P. and Smailagic, A. (1994). The CMU Mobile Computers as Maintenance Assistants. *NSF Workshop on Mobile and Wireless Information Systems*, Rutgers University, N.J., October.

Siewiorek, D., Smailagic, A., Bass, L., Siegel, J., Martin, R., and Bennington, B. (1998). Adranz: A Mobile Computing System for Maintenance and Collaboration. *Proc. IEEE International Conference on Wearable Computers*, Pittsburgh, PA, October 1998.

Siewiorek, D.P., Smailagic, A., Lee, J.C., and Tabatabai A.R.A (1994). An Interdisciplinary Concurrent Design Methodology as Applied to the Navigator Wearable Computer System. *Journal of Computer and Software Engineering 2*, No.2.

Smailagic, A. (1997). ISAAC: A voice activated speech response system for wearable computers. In *Proceedings of the IEEE International Conference on Wearable Computers*, Cambridge, MA.

Smailagic, A. and Martin, R. (1997). Metronaut: A Wearable Computer with Sensing and Global Communication Capabilities. *1st International Symposium on Wearable Computers*, Cambridge, Mass., October 13-14, 1997.

Smailagic, A. and Siewiorek, D.P. (1993). A Case Study in Embedded Systems Design: VuMan 2 Wearable Computer. *IEEE Design and Test of Computers 10,* No. 4.

Smailagic, A. and Siewiorek, D.P. (1994). The CMU Mobile Computers: A New Generation of Computer Systems. *Proceedings of COMPCON '94, IEEE Computer Society Press,* February Issue.

Smailagic, A. and Siewiorek, D. P. (1996). Modalities of Interaction with CMU Wearable Computers. *IEEE Personal Communications*, 14-25.

Smailagic, A. and Siewiorek, D. (1999). User - Centered Interdisciplinary Design of Wearable Computers. *ACM Mobile Computing and Communications Review 3*, 43-52.

Smailagic, A. and Siewiroek, D. (1999). System Level Design as Applied to CMU Wearable Computers. *Journal of VLSI Signal Processing Systems 21*, No. 3.

Smailagic, A., Siewiorek, D., Martin, R., and Reilly, D. (1999). CMU Wearable Computers for Real-Time Speech Translation. *Proc. 3rd International IEEE Symposium on Wearable Computers*, San Francisco, CA.

Smailagic, A., Siewiorek, D., Martin, R., and Stivoric, J. (1998). Very Rapid Prototyping of Wearable Computers: A Case Study of VuMan 3 Custom versus Off-the-Shelf Design Methodologies. *Journal on Design Automation for Embedded Systems 3*, 217-230

Smith, B., Bass, L., and Siegel, J. (1995). On site maintenance using a wearable computer system. In *Proceedings of the CHI '95: Human Factors in Computing Systems*, New York, NY, USA: ACM Press, 119-120.

Soitz, G. (1990). Target acquisition performance using a head-mounted cursor control device and a stylus with digitizing tablet. In *Proceedings of the Human Factors Society 34th Annual Meeting* (Santa Monica, CA), 405-409.

Spitzer, M. B., Rensing, N. M., McClelland, R., and Aquilino, P. (1997). Eyeglass-mounted displays for wearable computing. In *Proceedings of the International Symposium on Wearable Computers*, Los Alamitos, CA, USA: IEEE Computer Society, 48-51.

- lightweight functional eyeglass displays are possible and are being developed. They allow for see-through vision with reasonable acuity (a version has been developed at 320x240 pixels in a 5mm x 5mm display – the goal is 640x480). This display obptics are designed to project the image to 60 cm. [This is the MicroOptical Eyeglass Display].

Stute, N., Bautsch, H., Gdowski, M., Calhoun, C., Grigsby, S., Dixon, A., Cunningham, J., and Stautberg, D. (1998). *3D Graphics Investigation.* Final Technical Report (Contract No. F41624-97-D-5002 to AFRL/HESR).

Sunkpho, J., Garrett, J.H. Jr., Smailagic, A., and Siewiorek, D.P. (????). MIA: A Wearable Computer for Bridge Inspectors.

- Describes a wearable computing system that uses multimedia to aid bridge inspectors.

Tan, H. Z. and Pentland, A. (1997). Tactual Displays for Wearable Computing. In *Proceedings of the International Symposium on Wearable Computers*, Los Alamitos, CA, USA: IEEE Computer Society, 84-89.

Thomas, D.L. (1995). *Integrated Maintenance Information System (IMIS): User field demonstration and test: Final technical report.* Wright-Patterson AFB OH: Armstrong Laboratory, Human Resources Directorate, Logistics Research Division (AL/HR-TR-1995-0034).

Thomas, B., Tyerman, S., and Grimmer, K. (1997). Evaluation of three input mechanisms for wearable computers. *International Society for Wearable Computing Annual Meeting.*

- reports on an experiment comparing a wrist-worn keyboard, a virtual keyboard, and a Kordic keyboard for text entry tasks. Concludes that "the forearm keyboard is the best performer for accurate and efficient text entry" though they conclude that suitable design of the interface may improve performance with the other devices.

Thompson, C., Najjar, L. J., & Ockerman, J. J. (I997). Wearable computer based training and performance support systems. In S. Aronberg (Ed.), *19th Interservice/Industry Training, Simulation and Education Conference Proceedings.* Arlington, VA: National Training Systems Association, 746-752

- Used a wearable to perform a maintainence task utilizing collaboration with another user. Input could be via voice, wrist-worn keyboard, and/or hand-held pointing device. Key lessons learned: Wearables are most useful in confined hands-busy situations that do not allow use of traditional paper materials; new users require time to understand the features of a wearable and how best to use it; to maximize voice recognition accuracy, use acoustically distinct phrases; minimize the number of voice inputs by letting the computer automatically perform many user tasks; to improve usability of the voise input technique, develop applications that use a simple command-based user interface and minimize navigation.

Toub, R. (1996). *The Importance of Usability Studies and Experimental Design Evaluations for Wearable Computers.* (Online Report): Eugene, OR: University of Oregon, Department of Computer and Information Science.
(http://www.cs.uoregon.edu/~kortuem/htbin/download.cgi?/cs/www/home/research/wearables/Papers/usability.html)

Unger, D., Quill, L.L., and Masquelier, B.L. (1998). Fitness of Use Study of a Monocular Head Mounted Display: Usability Study. (Contract No. SPO900-94-D-0001). Dayton, OH: University of Dayton Research Institute.

- Technicians might use the system when performing infrequently performed tasks or tasks that require close coordination with what is written in the technical data (inspections and checklists). Specific recommendations are for very involved tasks: engine jobs, group tasks, tire changes, and inspections

Ware C. and Mikaelian, H.H. (1987). An evaluation of an eye tracker as a device for computer input. *Proceedings of Human Factors in Computing Systems and Graphics Interface Conference*: April 5-9, 1987, Toronto, Canada. Association for Computing Machinery (ACM). Ed. by John M. Carroll and Peter P. Tanner.

- "One of the most frequent operations which occurs in human computer interaction is the selection of an object displayed on a terminal. The item selected may be a word of text in a typical editing application or a menu item in a menu based interface. This is the 'pick' operation defined by Wallace (1976)." (Wallace, V.I. (1976) The semantics of graphic input devices. In Proceedings SIGGRAPH/SIGPLAN Conference. Computer Graphics, 10, 61-65.)

Williamson, D.T. (1999). Robust speech recognition interface to the electronic crewmember: Progress and Challenges. In J. Reising, R. Taylor & R. Onken (Eds.), *The Human-Electronic Crew: The Right Stuff?* Wright-Patterson AFB OH: Air Force Research Laboratory, Human Effectiveness Directorate, Crew System Interface Division (AFRL-HE-WP-TR-1999-0235), 115-121.

Whitefield, A. (1986). Human factors aspects of pointing as an input technique in interactive systems. *Applied Ergonomics 17*, 97-104.

## *REFERENCES*

[a] Thomas, D.L. (1995). Integrated Maintenance Information System (IMIS): User field demonstration and test: Final technical report. Wright-Patterson AFB OH: Armstrong Laboratory, Human Resources Directorate, Logistics Research Division (AL/HR-TR-1995-0034).

[b] Kancler, D.E., Revels, A.R., Quill, L.L. and Nemeth, K. (1999). SBIR Phase II: GMD Baseline Test Comparing the GMD with Traditional Displays (Contract No. F41624-98-C-5004, Project No. 6004). Dayton, OH: University of Dayton Research Institute.

[c] Revels, A.R., Kancler, D.E., Quill, L.L. and Nemeth, K. (1999). SBIR Phase II: Test #2, GMD Mobility Test Using the GMD in a Synthetic Mobile Environment (Contract No. F41624-98-C-5004, Project No. 6004). Dayton, OH: University of Dayton Research Institute.

[d] Stute, N., Bautsch, H., Gdowski, M., Calhoun, C., Grigsby, S., Dixon, A., Cunningham, J., and Stautberg, D. (1998). 3D Graphics Investigation. Final Technical Report (Contract No. F41624-97-D-5002 to AFRL/HESR).

[e] Chapman, D. and Simmons, J. (1995). A comparative evaluation of voice versus keypad input for manipulating electronic technical data for flightline maintenance technicians. Master's Thesis, Air Force Institute of Technology, Wright-Patterson, AFB, Ohio, 1995.

[f] McMillan G.R., Calhoun G.L., Masquelier B.L., Grigsby S.S., Quill L.L., Kancler D.E., Nemeth K., and Revels A.R. (1999). Comparison of hands-free versus conventional wearable computer control for maintenance applications. Human Factors and Ergonomics Society Annual Meeting '99.

[g] Borah, J. (1989) *Helmet-Mounted Eye Tracking for Panoramic Display Systems – Volume I: Review of Current Eye Movement Measurement Technology* (AAMRL-TR-89-019) (NTIS: AD-A273 101/6/XAB).

[h] Ware C. and Mikaelian, H.H. (1987) An evaluation of an eye tracker as a device for computer input. *Proceedings of Human Factors in Computing Systems and Graphics Interface Conference*: April 5-9, 1987, Toronto, Canada. Association for Computing Machinery (ACM). Ed. by John M. Carroll and Peter P. Tanner.

[i] Borah, J. (1995). *Investigation of Eye and Head Controlled Cursor Positioning Techniques.* AL Tech. Report #AL/CF-SR-1995-0018

[j] Liggett, K.K., Kustra, T.W., Reising, J.M., and Hartsock, D.C. (1997). A comparison of touch, manual, and head controllers for cursor movement. *Proceedings of the Human Factors and Ergonomics Society 41st Annual meeting*, 80-84.

[k] Anderson, T.R. (1998). The technology of speech-based control. In *Alternative Control Technologies: Human Factors Issues.* RTO Lecture Series 215 (RTO-EN-3 AC/323(HFM)TP/1).

[l] Borah, J. (1998). Technology and application of head based control. In *Alternative Control Technologies: Human Factors Issues.* RTO Lecture Series 215 (RTO-EN-3 AC/323(HFM)TP/1).

[m] McMillan, G.R., Eggleston, R.G., and Anderson, T.R. (1997). Nonconventional controls. In G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics* (2nd ed.). New York: John Wiley & Sons.

[n] Lin, M.L., Radwin, R.G., and Vanderheiden, G.C. (1992). Gain effects on performance using a head-controlled computer input device. *Ergonomics 35*, 159-175.

[o] Thomas, B., Tyerman, S., and Grimmer, K. (1997). Evaluation of three input mechanisms for wearable computers. *International Society for Wearable Computing Annual Meeting.*